```
/* A Secure Authentication and Key Management Protocol using ECC
for Deployment of Internet of Vehicles (IoV) Concerning Smart Cities */
/*SPDL Specfication*/
usertype Vnid,Nu,Tsu,Csid,RSUid,Fsid,Tsfs,Nfs,Tsu,Nu,
VN,IDu,TScs,Ncs;
const puk:Function;
secret prk:Function;
inversekeys (puk,prk);
usertype TimeStamp,Authsuccess,Authfail;
hashfunction H1,H2,H3,H4;
protocol AKAP(U,FS,RSU,CS,TA)
{
role U
{
const Kcib:SessionKey;
send_1(U,FS,{{VNid,Nu,TSu}prk(U)}puk(FS));
read_2(FS,U,{{CSid,FSid,TSfs,Nfs,TSu,Nu,VN}prk(FS)}puk(U));
read_8(FS,U,{{VNid,RSUid,Authsuccess,Authfail}prk(FS)}puk(U));
claim_U1(U,Secret,Kcib);
claim_U2(U,Secret,Nu);
claim_U3(U,Secret,VN);
claim_U4(U,Secret,Nfs);
claim_U5(U,Niagree);
claim_U6 (U,Nisynch);
}
role FS
{
const SKfc,SKcf:SessionKey;
read_1(U,FS,{{VNid,Nu,TSu}prk(U)}puk(FS));
send_2(FS,U,{{CSid,FSid,TSfs,Nfs,TSu,Nu,VN}prk(FS)}puk(U));
send_3(FS,CS,{H1(SKfc),(FSid,IDu,Nfs,TSfs,VN)}SKfc);
read_6(CS,FS, {H4(SKcf),(VNid,RSUid,Authsuccess,Authfail)}SKcf);
send_7(FS,RSU,
{{VNid,RSUid,Authsuccess,Authfail}prk(FS)}puk(RSU));
send_8(FS,U,{{VNid,RSUid,Authsuccess,Authfail}prk(FS)}puk(U));
claim_FS1(FS,Secret,SKfc);
claim_FS2(FS,Secret,Nfs);
claim_FS3(FS,Secret,VN);
claim_FS4(FS,Niagree);
claim_FS5(FS,Nisynch);
}
```

```
role CS
{
const SKfc,SKct,SKcf,SKtc:SessionKey;
read_3(FS,CS,{H1(SKfc),(FSid,IDu,Nfs,TSfs,VN)}SKfc);
send_4(CS,TA, {H2(SKct),(CSid,FSid,TScs,Ncs,VN)}SKct);
read_5(TA,CS, {H3(SKtc),(VNid,RSUid,Authsuccess,Authfail)}SKtc);
send_6(CS,FS, {H4(SKcf),(VNid,RSUid,Authsuccess,Authfail)}SKcf);
claim_CS1(CS,Secret,SKct);
claim_CS2(CS,Secret,SKcf);
claim_CS3(CS,Secret,VN);
claim_CS4(CS,Niagree);
claim_CS5(CS,Nisynch);
}
role TA
{
const SKtc,SKct:SessionKey;
read_4(CS,TA, {H2(SKct),(CSid,FSid,TScs,Ncs,VN)}SKct);
send_5(TA,CS, {H3(SKtc),(VNid,RSUid,Authsuccess,Authfail)}SKtc);
claim_TA1(TA,Secret,SKtc);
claim_TA2(TA,Secret,SKct);
claim_TA3(TA,Secret,TScs);
claim_TA4(TA,Niagree);
claim_TA5(TA,Nisynch);
}
role RSU
{
read_7(FS,RSU,
{{VNid,RSUid,Authsuccess,Authfail}prk(FS)}puk(RSU));
claim_RSU(RSU,Secret,VN);
claim_RSU(RSU,Niagree);
claim_RSU(RSU,Nisynch);
}
}
```