1. [20 points]

Protocol4(A, B) {
    chan ← [];
    hst ← []; // connection history
    mKey ← random();
    startSystem(A, Client4(A,B,mKey));
    startSystem(B, Server4(B,A,mKey));
    startSystem(Attacker());
}

Attacker() {
    <read chan>
}

Client4(A, B, mKey) {
    // atomicity points: 1
    nL ← 0;
    while (true) {
        nL ← nL + 1;
        tx([A,B,nL]);
        1: msg ← rx([B,A,.,.]);
        if (msg[3] = enc(mKey,nL)) {
            nR ← msg[2];
            hst.append([A,nL,nR]);
            tx([A,B,nL,enc(mKey,nR)]);
        }
    }
}

Server(B, A, mKey) {
    // atomicity points: 1,2
    nL ← 0;
    while (true) {
        1: msg ← rx([A,B,.,.]);
        nR ← msg[2];
        nL ← nL + 1;
        tx([B,A,nL,enc(mKey,nR)]);
        2: msg ← rx([A,B,.,.]);
            hst.append([B,nL,nR]);
        }
    }
}

For each assertion below, prove or disprove whether the assertion holds for Protocol4. If you prove, present an invariantly-complete predicate that implies the assertion’s predicate. If you disprove, present a counter-example evolution.

a. Inv (mKey ncf α)

b. Inv forall(i in hst.keys: hst[i] = [B,nB,nA] ⇒ [A,nA,nB] in hst[0..i-1])
2. [10 points]

Repeat problem 1 but now with an attacker that can read and write chan.
An organization has a PKI (public-key infrastructure) for its users consisting of a single certification authority (CA) and a single directory server (DS), which any user can contact to obtain certificates and CRLs. Certificates have an expiry time of 1 year. CRLs are issued hourly. Answer the following questions. Be brief and precise.

a. Describe the steps taken when a user $A$ joins the organization.

b. User $C$ steals user $A$’s private key and $A$ does not realize this. How long after this can $C$ impersonate $A$, i.e., talk to a user $B$ and convince $B$ that it is talking to $A$.

c. User $C$ steals user $A$’s private key and $A$ realizes this.
   – Describe the steps $A$ takes.
   – How long after these steps can $C$ impersonate $A$. 
The program below uses the Diffie-Hellman protocol with public parameters g and p.

```c
Protocol5(A, B, g, p) {
    chan ← [];
    startSystem(A, Client5(A,B,g,p));
    startSystem(B, Server5(B,A,g,p));
    startSystem(Attacker());
}

Attacker() {
    ....
}
```

```c
Client5(A, B, g, p) {
    // atomicity points: 1
    nL ← random();
    tL ← g^nL mod p;
    tx([A,B,tL]);
    1: msg ← rx([B,A,.]);
    tr ← msg[2];
    keyDH ← tr^nL mod p;
    data ← random();
    tx([A,B,'DATA',enc(keyDH,data)]);
}
```

```c
Server5(B, A, g, p) {
    // atomicity points: 1,2
    1: msg ← rx([A,B,.]);
    tr ← msg[2];
    nL ← random();
    keyDH ← tr^nL mod p;
    tx([B,A,tL]);
    2: msg ← rx([A,B,'DATA',.]);
    data ← msg[3];
}
```

a. Can an attacker who can only read chan obtain data?
a. Can an attacker who can read and write chan obtain data?

In each part above, answer yes or no. If you answer yes, give an evolution ending in a state where the attacker has data. If you answer no, explain briefly (no need for predicates).