Exercise 4

Exercises Peer-to-Peer-Systems and Security (SS2010)

Monday 14.6 2010
Hand-in: Monday 28.6. 2010 in lecture or per mail
Exercise: to be announced

Dipl.-Inform. Heiko Niedermayer
Lehrstuhl für Netzarchitekturen und Netzdienste
Technische Universität München

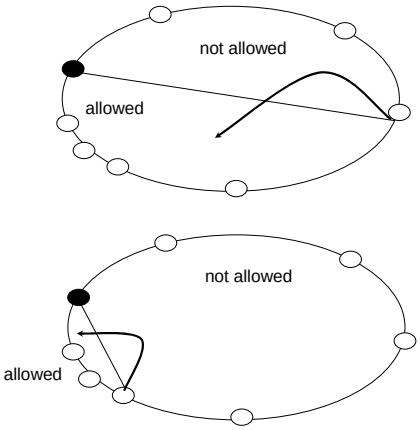
Rules: There will be five exercise sheets with each 10 points. You have to achieve 50 % of the points and present a solution in the exercise course to get the 0.3 bonus.

Task 1 (2 Points) Eclipse Attack
In this task we discuss node and data eclipse.

a) It is possible in Chord to eclipse a node for messages to the node with one other node. Where do you need to position yourself as attacker and describe why this is the case?
b) Why is data eclipse trivial in standard DHTs like Chord (in the lecture we have seen that it is not easy in the KAD network)?

Solution:
a)
If the attacker positioned itself as predecessor of the node, all messages to the attacked node go through the predecessor.

The reason is that the predecessor is responsible for the interval before the node and the routing does not allow to route to an ID that is larger than the ID of the node. Thus, one can only route through preceeding IDs.



The image tries to display the situation, one can only route to closer nodes yet with „smaller“ preceeding IDs to the target. In case of item lookups, only the predecessor can decide if there is no further other node between an ID of the item and the corresponding node responsible for the item.

b)
The logical position of a node in the DHT can be affected easily by an attacker. So they can simple selected the needed position as ID in the DHT (some security proposals: ID = sha(IP), then this may require an appropriate IP address or similar measures are taken to make the ID assignment less free, yet this is usually not checked).

Task 2 (2 Points) Authentication

In this task we specify a cryptographic protocol which is meant to be used for mutual authentication.

a) Specify on what information and when in the protocol do the entities A,B, and S detect a successful authentication run?
b) The protocol is insecure. Find an attack. (The strength of the attacker is that it can read, send, fake, and drop messages in the network, yet it cannot break cryptography. This is a common security model in network security.)

Protocol for Task 2:

Prerequisites:
S is a TTP.
Each participant X has a shared key with S. This key is called k kXS.
Let kab = Nb.
Protocol:

```
A -> B: Na, A
B -> S: Na, A, B,{Nb}kBS
S -> A: B, {Na, Nb}kAS
A -> B: {Nb}kab
```

Solution:

A recognizes B: in step 3, the server sends a message which contains Bob and Na
B recognizes A: in step 4 by receiving Nb from A which was protected with the shared key kab
S does not detect the success of a run. It might recognize B on the knowledge of the kBS encrypted and integrity protected). It does not recognize A, but provides information on Nb that only A should be able to read.

Attack:

- The attacker replays an encrypted nonce from a previous interaction of the attacker C_B as normal entity C.

```
1. C -> B: Nc,C
2. B -> S: Nc,C,B,{Nb}kBS
3. S -> C: B,{Nc,Nb}kCS
4. C -> B: {Nb}kcb
Now Alice(A) wants to communicate with Bob(B).
Attacker C_B is sitting in between.
1. A -> C_B: Na,A
2. C_B -> S: Na,A,B,{Nb}kBS
3. S -> A  : B, {Na,Nb}kAS
4. A -> C_B: {Nb}kab
 => A recognized C_B as Bob, C_B also knows kab and
can know communicate with Alice as Bob.
```

Task 3 (2 Points) Authentication

Do the same as in task 2, yet with a different protocol. Hint: Use information from previous runs to attack the protocol. Sig_X stands for encryption with private key.

Prerequisites: S is TTP and for each participant X S knows the corresponding public key PK_X. All participants know the public key PK_S of S.
Let kab = hash(Na,Nb).
Protocol:

```
A -> S: A, Enc_PK_S(Na, B)
S -> A: PK_B, Enc_PK_A(Sig_S(Na, A))
A -> B: Enc_PK_B(Na, A, B, Sig_S(Na, A))
B -> A: Nb, {Na}kab
A -> B: {Nb}kab
```

Solution:

B recognizes A in the 5th message due to the signature from the server in message 3 and the knowledge of nb and kab.
A recognizes B in the 4th message due to the knowledge of Na and kab.
S does not detect success of the protocol run. It does not recognize A, but only provides information that only A can read, its signature of Na and A encrypted with A's public key.

Attack:

- The attacker C_A uses a previous communication of A and C_A to attack A and B.

A communicates with C (later to become C_A).

```
1. A -> S: A, Enc_PK_S(Na,C)
2. S -> A: PK_C,Enc_PK_A(Sig_S(Na,A))
3. A -> C: Enc_PK_C(Na,A,C,Sig_S(Na,A))
4. C -> A: Nc,{Na}kab
5. A -> C: {Nc}kab
```

Now C attacks A and B by impersonating A. Thus, we will call C now C_A. It reuses the old nonce Na and signature from the server that it received in the 3rd message. C_A knows the public key PK_B of B (if not, it could ask S and receives it in protocol step 2).

```
3. C_A -> B: Enc_PK_B(Na,A,B,Sig_S(Na,A))
4. B -> C_A: Nb,{Na}kab
5. C_A -> B: {Nb}kab
```

=> B accepts C_A as A and C_A knows kab.

Task 4 (2 Points) Some questions

Answer the questions with knowledge from the lecture.

a) Cryptographic identities seem to make authentication a lot easier. Let assume, we use cryptographic identities. Do we still need a Certificate Authority? If yes, for what? If no, why not?
b) Why is trust important for key distribution?
c) Why does Zfone or SSH in the Baby Duck model not simply use a conventional authentication protocol to authenticate the communication partners? What problem do they try to resolve?

Solution:

a)

Yes, since anyone can create a cryptographic identity (public key). It is not clear who (true identity) this entity is or what kind of access rights it has.

If it is sufficient to recognize a user again or to fight a man-in-the-middle attack between IDs, then a CA is not necessary.

b)

Since the knowledge of a key represents a way to identify entities and secure communication, one trusts that the other legitimate communication partner handles his keys in a responsible way.

If external entities like TTP or CA are used, one needs to trust their operation, that they send and forward the correct information. A CA or TTP is also used to connect key and identity for an entity. This can only happen if the entity can trust into the operation and accurateness of the CA or TTP. If this is not the case, one has to assume that maybe the key of the wrong person is sent or that confidentiality is not ensured.

c)

They assume that there is no global trustworthy TTP or CA, at least not one that they can use in all cases or for all application. Since security is almost impossible for the first contact without external help of CA or TTP, these protocols try to resolve the corresponding problems with other methods.

Task 5 (2 Points) Sybil Attack

In the lecture, we introduced the sybil attack.

a) Why is the sybil attack fatal for the security of a Peer-to-Peer network?
b) Does requiring an email address prevent nodes from launching a sybil attack?
c) How could an attacker profit from a sybil attack in an auction site with a reputation system like Ebay?

Solution:

a)

A lot of assumptions about the good-naturedness or resilience of a P2P network are about having a diverse set of nodes. A sybil attack reduces diversity. If a single node controls large parts of the network, it gains a lot of power and becomes a single-point of failure. This power can be used to attack the network and also measures for security, that might be based on voting.

b)

No, since one can easily generate multiple email addresses.

c)

An attacker can have multiple accounts, e.g. non-existing neighbors or room mates. As seller it may be sufficent to use addresses that are not true as one does not need to be reachable.

The sybil attacker can:

- Increase the price by bidding with a sybil in its own auctions (only costs some dues if the attacker won the bidding for its sale)
- The attacker can also give good ratings to its own nodes and undermine the reputation system in this way
- He could destroy reputation of his competitors due to his anonymity