Open Book Examination
Introduction to Computer Security
215010

University of Twente
department of computer science

October 26th, 2009

Answer 6 out of 8 questions. Each question is worth 15%.

Illegible answers or excessively long answers will not be marked.

1. Prove the following properties of El Gamal encryption $E_k(.)$ and decryption $D_{k'}(.)$, where k, k' form a key pair:
   (a) El Gamal is correct, i.e. for all m : $D_{k'}(E_k(m)) = m$
   (b) El Gamal encryption is a privacy homomorphism, i.e. for all m, m' and suitable operators $\otimes,\times$: $E_k(m) \otimes E_k(m') = E_k(m \times m')$

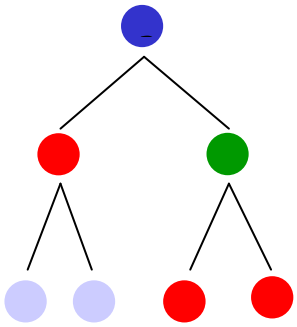
2. General questions about Biometrics.
   (a) Define the FAR, FRR, and EER in a biometric system.
   (b) Explain why the EER of a biometric system cannot be 0.

3. Which methods (e.g. keylogger) can an attacker use, and how, to get hold of the password or pin code? Indicate which attack is most likely to happen and why.
   (a) Desktop PC in an Internet Café: password/pincode
   (b) iPhone or iPod: pin code
   (c) Fixed line phone: pin code
   (d) Web browser with on screen keyboard: password/pin code
   (e) Voice controlled computer: password/pin code
   (f) ATM: pin code

4. Questions about the "School of Phish" experiment.
   (a) The participants are self selected. What do you think this means for the outcome of the experiments described in the paper?
   (b) Creating a clone of a web page may cause a copyright violation. Can you think of at least two methods in which researchers can avoid this problem?

5. General questions about Hash chains.
   (a) What is a hash chain
   (b) Why should a hash chain be generated in its entirety before it can be used?
   (c) Can you think of a way of mitigating this disadvantage by using a hash chain of hash chains?

6. Consider the searching in encrypted data method of Brinkman. Give an interactive client-server query protocol consisting of a recursive algorithm for the client and a simple request response loop for the server that:
   - Minimizes the amount of storage for the client
   - Does not search sub trees that cannot satisfy the query
   - You may use the sample tree to the right to illustrate your protocol



7. Suppose six people want to communicate securely with each other, such that the communication of none of the possible pairs of people can be eavesdropped by the remaining four persons.
   (a) If they use a symmetric cipher, how many keys would they need in total?
   (b) If they use an asymmetric cipher, how many keys would they then need in total?

1. $A \rightarrow S$: $\{N_a, B\}pk(S)$
2. $S \rightarrow A$: $\{N_a, K_{ab}\}pk(A)$
3. $S \rightarrow B$: $\{K_{ab}\}pk(B)$
4. $B \rightarrow A$: $\{B\}K_{ab}$

8. The protocol above aims at key distribution of K_{ab}. Here N_a is a nonce, $pk(X)$ is the public key of agent X, and $\{M\}pk(X)$ is message M encrypted with the public key of X using a suitable public-key algorithm (messages 1, 2 and 3). Message 4 uses symmetric encryption with key K_{ab}.

   To ensure the delivery of the key to the correct agents, authentication is required. Explain for each of the six options below whether the protocol aims at:
   (a) Mutual authentication without the use of a server
   (b) Mutual authentication using a trusted third party (TTP)
   (c) Provide authentication of A to S
   (d) Provide authentication of S to A
   (e) Provide authentication of B to A
   (f) Provide authentication of B to S

Client:

```
seed = …
query = …
root = 1
Search(root, query)

procedure Search(id, query)
        poly = Genpoly(Rand(id, seed))
        left=Eval(poly,query)
        Send(server, (id, query))
        (right, n) = Receive (server)
        if left+right = 0 then
                for i = 1 to n do
                        Search(id.i, query)
                od
        fi
end
```

Server:

```
tree = 1 (p1,    1.1(p2),
                 1.2(p3,1.2.1(p4),1.2.2(p5),1.2.3(p6))))

Function Getpoly(… id(p,c1..cn)…, id) = p
Function Getnum(… id(p,c1..cn)…, id) = n

loop
        (id, query) = Receive(client)
        right = Eval(Getpoly(tree,id), query)
        n = Getnum(tree,id)
        Send(client, (right, n) )
end
```