Security on the Internet, winter 2008

MICHAEL NÜSKEN, DANIEL LOEBENBERGER

8. Exercise sheet
Hand in solutions until
Monday, 5 January 2009, 11^{59}am (deadline!).

As usual: Any claim needs a proof or an argument.

Exercise 8.1 (Key exchange threats). (15 points)

We have considered the Diffie-Hellman key exchange: Given a group G consisting of powers of a generator g of order ℓ, so $G = \{1, g, g^2, \ldots, g^{\ell-1}\}$ such that the discrete log problem is difficult, ie. given $h \in G$ there is no efficient (ie. randomized polynomial time) algorithm to determine η with $h = g^\eta$. To fix a shared secret key, Alice sends g^α and Bob sends g^β. Then both can compute the shared key $g^{\alpha\beta}$.

Protocol DH. Diffie-Hellman key exchange.

1. Alice chooses $\alpha \in \mathbb{N}_{<\ell}$ and computes g^α.
2. Bob chooses $\beta \in \mathbb{N}_{<\ell}$ and computes g^β.
3. Alice computes $(g^\beta)^\alpha = g^{\alpha\beta}$.
4. Bob computes $(g^\alpha)^\beta = g^{\alpha\beta}$.

$$\xrightarrow{\qquad g^\alpha \qquad}$$
$$\xleftarrow{\qquad g^\beta \qquad}$$

Now both can use $g^{\alpha\beta}$ to derive common secrets for the subsequent message exchanges. What if Wilma puts herself in the middle? She will have a common secret $g^{\alpha\omega}$ with Alice and a common secret $g^{\omega'\beta}$ with Bob, and as long as she continues to pass all messages on, neither Bob nor Alice will notice anyhting apart possibly from a slighlty slower connection. So we modify this.

Protocol DH+sign+ack. Signed and acknowledged Diffie-Hellman key exchange.

1. Alice chooses $\alpha \in \mathbb{N}_{<\ell}$, computes g^α and signs ['Alice', g^α].
2. Bob chooses $\beta \in \mathbb{N}_{<\ell}$, computes g^β and signs ['Bob', g^β].
3. Alice computes $(g^\beta)^\alpha = g^{\alpha\beta}$ and a hash.
4. Bob computes $(g^\alpha)^\beta = g^{\alpha\beta}$ and a hash.

$$\xrightarrow{\qquad [\text{'Alice'}, g^\alpha]_{\text{Alice}} \qquad}$$
$$\xleftarrow{\qquad [\text{'Bob'}, g^\beta]_{\text{Bob}} \qquad}$$
$$\xrightarrow{\qquad \text{hash}(0, g^{\alpha\beta}) \qquad}$$
$$\xleftarrow{\qquad \text{hash}(1, g^{\alpha\beta}) \qquad}$$

Sorry, we forgot to be polite. We should first say Hello, shouldn't we?

Protocol DH+polite+cookie. Polite Diffie-Hellman key exchange with a cookie.

1. Alice wants to talk. $\xrightarrow{\text{I want to talk}}$
2. Bob agrees and chooses a cookie c, which is a suitably random number, for example, the hash value of *her* IP address and some fixed secret of Bob. (It's nice if the number is deterministically determined!) $\xleftarrow{\text{Ok, I listen for cookie } c.}$
3. Alice chooses $\alpha \in \mathbb{N}_{<\ell}$, computes g^{α} and signs $[\text{'Alice'}, g^{\alpha}]$. $\xrightarrow{c,\ [\text{'Alice'}, g^{\alpha}]_{\text{Alice}}}$
4. Bob chooses $\beta \in \mathbb{N}_{<\ell}$, computes g^{β} and signs $[\text{'Bob'}, g^{\beta}]$.
5. Bob computes $(g^{\alpha})^{\beta} = g^{\alpha\beta}$ and a hash. $\xleftarrow{[\text{'Bob'}, g^{\beta}]_{\text{Bob}},\ \text{hash}(1, g^{\alpha\beta})}$
6. Alice computes $(g^{\beta})^{\alpha} = g^{\alpha\beta}$ and a hash. $\xrightarrow{\text{hash}(0, g^{\alpha\beta})}$

Here is a further polite variant.

Protocol DH-IPsec. Modified Diffie-Hellman key exchange.

1. Alice chooses $\alpha \in \mathbb{N}_{<\ell}$, computes g^{α}. $\xrightarrow{\text{I want to talk, } g^{\alpha}.}$
2. Bob chooses $\beta \in \mathbb{N}_{<\ell}$, computes g^{β}. $\xleftarrow{\text{Ok, } g^{\beta}.}$
3. Alice computes $(g^{\beta})^{\alpha} = g^{\alpha\beta}$ and uses it to encrypt her name and a signature to her share g^{α}. $\xrightarrow{E_{g^{\alpha\beta}}(\text{'Alice'}, [g^{\alpha}]_{\text{Alice}})}$
4. Bobd computes $(g^{\alpha})^{\beta} = g^{\alpha\beta}$ and uses it to encrypt his name and a signature to his share g^{β}. $\xrightarrow{E_{g^{\alpha\beta}}(\text{'Bob'}, [g^{\beta}]_{\text{Bob}})}$

Consider each of the above protocols in the following questions. (Be brief, but don't forget the essential arguments.)

$\boxed{2}$ (i) *Woman in the middle*: Try to put Wilma in the middle. What happens?

$\boxed{2}$ (ii) *Mutual authentication*: Examine which of the given protocols ensure that Alice' partner is Bob.

$\boxed{2}$ (iii) *Perfect Forward Security*: Next, suppose that the Beagle Boys intercepted the conversation between Alice and Bob. Then after the conversation is terminated the Beagle Boys take over Alice' and Bob's entire equipment including their secret keys. Will they be able to read what Alice and Bob told each other?

$\boxed{2}$ (iv) *Denial of Service*: Daniel is a weird person that only wants to prevent say Bobs' computer to do good work. So he floods Bob with tons of requests. For each of these requests Bob's computer is forced to compute and send an answer. Consider vaguely the effort which Daniel and Bob have to spend for their first messages and vote for the 'best' protocol.

$\boxed{2}$ (v) *Endpoint Identifier Hiding*: Eve does not want to be spotted, so she only listens on the conversation. If she can detect who the partners are, this is already valuable information for her. Which protocols hide the identity of Alice and/or Bob?

(vi) *Live Partner Reassurance*: Romeo likes repetions and so after listening to a conversation, he calls Bob with replayed messages from the overheared talk making him think he is Alice. (Imagine this could be successfully done when you log in to your home banking account!) Examine the given protocols under this attack. $\boxed{2}$

(vii) Devise a protocol that Romeo cannot trick. (Do not forget to argue!) $\boxed{3}$

(viii*) Devise a protocol that is not vulnerable to any of these attacks. $\boxed{+0}$

Exercise 8.2 (IKE). (8 points)

(i) Read the RFC 2409 on IKE as well as the RFC 4306 on IKEv2 and describe briefly the major differences. $\boxed{4}$

(ii) Show how someone who knows both Alice' and Bob's public encryption key (and neither side's private key) can construct an entire IKE exchange based on public encryption keys that appears to be between Alice and Bob. $\boxed{2}$

(iii) Design a protocol in which one side has a public signature key and the other side has a public encryption key. $\boxed{2}$

Exercise 8.3 (IPsec in practice). (0+4 points)

Which (common) applications do use/implement IPsec? $\boxed{+4}$

Where is it used in our vicinity? (Where within b-it, computer science Bonn, computer science Aachen, University of Bonn, University of Aachen? Which services there do use it?)