Exercise 4.1 (Project, part 1). (16+12 points)

Choose either IPsec, ssh/scp or Signal protocol (OTR, Signal, Whatsapp) for this exercise. Make your choice public via https://doodle.com/poll/ggap4yteb6mtukec.

Find sources that describe the chosen protocol and study them. These sources should include the relevant up-to-date RFCs if any.

(i) Supply a list of all used sources!  

(ii) Give a short description of the protocol (in your own words!), enough to answer the following questions.

(iii) Where is the chosen protocol (typically) located in the OSI-model? What are pros and cons of this placement?

(iv) How is the start of a communication specified and how is the key exchange done in the chosen protocol? Is a man-in-the-middle attack possible? Does the key exchange include cipher negotiation, if so how?

(v) How is the data transfer secured? Which authenticated encryption schemes are allowed?

(vi) Discuss! +12

Exercise 4.2 (Key exchange threats). (9 points)

Consider the following protocols for establishing shared keys. We are about to discuss some aspects of security in an informal way.

Assume there is an infrastructure such that Alice and Bob have access to any party’s true public key.

For ease of notation, \([m]_{\text{Alice}}\) denotes the pair consisting of the message \(m\) and a valid public-key signature of \(m\) produced by Alice. Similarly \(\{m\}_K\) shall denote the message \(m\) symmetrically authenticated and encrypted by \(K\).

For Protocol 2 and Protocol 4 let \(\pi = (G, g, q)\) be group parameters. We assume that the discrete logarithm problem relative \(\pi\) is suitably hard. Let \(h\) be a collision resistant hash function. For Protocol 4 Alice has a secret passphrase \(a\) and her public key is \(A = g^{h(a)s}\), where \(s\) is a random number (the “salt”).
Protocol 2.
1. Alice chooses $a \in \mathbb{N}_{<\#G}$, computes $g^a$ and signs $['Alice', g^a]$. $\rightarrow [Alice', g^a]_{Alice}$
2. Bob chooses $b \in \mathbb{N}_{<\#G}$, computes $g^b$ and signs $['Bob', g^b]$. $\leftarrow [Bob', g^b]_{Bob}$
3. Alice verifies the incoming message and computes $(g^b)^a = g^{ab}$ and a hash. $h(0|g^{ab})$ $\rightarrow$
4. Bob verifies the incoming message and computes $(g^a)^b = g^{ab}$ and a hash. $h(1|g^{ab})$ $\leftarrow$

Protocol 3.
1. Alice wants to talk. $\rightarrow$ I want to talk
2. Bob agrees and chooses a cookie $c$, which is a suitably random number, for example, the hash value of Alice’s IP address and some fixed secret of Bob. (It’s nice if the number is deterministically determined!) $\leftarrow$ Ok, I listen for cookie $c$.
3. Alice computes RSA keys $(N, e)$ and $(N, d)$. $\rightarrow$ $enc_{(e, N)}(pad(K))$
4. Bob chooses a 128-bit number $K$, encrypts $K$ with Alice’s RSA key $(N, e)$ with a secure padding scheme. $\leftarrow$

Protocol 4.
1. Alice wants to talk. $\rightarrow$ Hello, I am Alice.
2. Bob chooses $b \in \mathbb{N}_{<d}$ and computes $B = g^b$, $K = A^b$, and $h(K)$. $\rightarrow$ Ok, $B, \{h(0|K)\}_K$
3. Alice computes $K = B^{h(a|s)}$, decrypts the last message and checks whether she computes the same values $h(0|K)$. $\leftarrow$ $\{h(1|K)\}_K$
4. Bob checks whether he computes the same value $h(1|K)$.

Consider each protocol in the following questions. Be brief, but don’t forget the essential arguments.

(i) **Person in the middle**: Peter puts himself in the middle. What happens?

(ii) **Perfect Forward Security**: Next, suppose that the Beagle Boys intercepted the conversation between Alice and Bob. Then, after the conversation is terminated and all ephemeral data is shredded, the Beagle Boys take over Alice’ and Bob’s entire equipment including their long-term secret keys. Will they be able to read what Alice and Bob told each other?

(iii) **Live Partner Reassurance**: Raoul likes repetitions and so after listening to a conversation, he calls Bob (or is called by Alice) with replayed messages from the overheard talk making him (her) think she (he) is Alice (Bob). Examine the given protocols under this attack.