PSL EX.2. (OFMC) Preliminary Draft

2.1. The initial step in the protocol requires A to generate a nonce and transmit it to B. Upon reception, B generates a new nonce and appends it to the nonce sent from A. Subsequently, B uses the key derivation function to generate a new key by inputting the shared key and the nonces. Furthermore, B generates a mac for the nonces with the new key. This mac, and the nonce B generated, is sent to A. Likewise, A can compute the new key from the key derivation function by following the similar steps as B. Moreover, A can compute a mac in the same manner as B and compare it to the one sent by B. If the mac's are similar to each other, A is able to authenticate B because to the get a match, B must have used the new secret key to generated the mac. In the subsequent step, B receives a mac from A for nonces, shared number between the two entities, and the previous mac. Since this mac is also generated using the shared key, B is able to authenticate A by computing a similar mac with the same arguments.
So in conclusion, the initial key is the one already shared by A and B. Nonces and the initial key are used to generate a new key and a mac is then created for the nonces using the new key. Upon reception, both entities can compute a mac using similar arguments known to both of them. Comparing the mac at each stage, therefore, proves the authenticity of either entity and it can also be assumed that the message itself has not been tampered. (Because a completely different mac would be generated for a different message).

2.2. In the first session, the intruder is impersonating B and receives a nonce from A. In the same session, the intruder sends back the nonce to A. In the next session, A instantiates B and “thinks” that this is the initial message transmitted by A and, therefore, generates another nonce and sends it to the intruder along with the hashed mac of the newly generated key and the nonces. The goal broken here is that weak authentication is violated in the protocol because in any session, one of the entity is not involved at any moment in the communication. The violation is represented in the statement as ”if there is a request fact without a matching witness fact” and this is reflected in the attack trace when A finishes the protocol without including B. [pg. 10 OFMC documentation]

2.3. In the initial exchange of messages in the communication, the nonce is not encrypted and it is vulnerable upon transmission on an insecure channel. Moreover, B doesn't know about the identity of A in the first transmission of the message. A could include its identity along with the nonce in a message and encrypt it using the shared secret key between A and B which is: exp(exp(g,secretk(A)),secretk(B)). Furthermore, B could also encrypt the nonce it generates using the shared key in the second transmission of the message. Furthermore, OFMC verifies the prevention of attacks.

2.4. With the secret key of A, the intruder can generate the secret key shared by only A and B. So altering the protocol from the previous step would serve no purpose because intruder can access any data in transit between A and B that is encrypted with the now exposed key. Perfect forward secrecy [look into it]