Homework Assignment 7

Due on Wednesday, November 4, 2020

1 ElGamal Authentication (6 points)

Once Happy understood ElGamal signatures, he was excited to use them for authentication. He wants to send an authenticated message m to Bob so that Bob can verify that m came from him.

Happy has an ElGamal signing key (g, p, x) and Bob has the corresponding verification key (g, p, a). We denote the signing algorithm using that key pair by S and the verification algorithm by V. Happy and Bob also have a cryptographic hash function h whose output is the same length as the signatures produced by S.

Here's Happy's idea. Bob sends him a fresh tag r. Happy signs r and attaches it to a hash of his message. Bob checks the tag's signature and accepts the message.

	Happy		Bob
1.		$\xleftarrow{r}$	Choose random string r.
2.	Compute $s = S(r) \oplus h(m \oplus r)$	$\xrightarrow{(m,s)}$	Check $V(r, s \oplus h(m \oplus r))$. If check succeeds, accept m as coming from Happy.

Questions

1. Verify that Bob accepts every message that Happy sends in this way (assuming no errors in transmission). Explain.

2. Mallory wants to replace m with a message m' of his choosing and get Bob to accept it as valid. Describe in detail how he can do this. Assume that Mallory is carrying out a man-in-the-middle attack, but she does not know Happy's signing key, cannot forge signatures $S(x)$ for messages x of Mallory's choosing, and has no knowledge of r, m, and s before she sees them coming over the channel.

3. Suggest a way to fix this protocol to thwart Mallory's attack. Your suggestion should not use any more rounds of communication nor assume any other encryption system or secret keys. Explain.

   [Hint: Think about a better way to use h to "bind" m to the signature.]

2 Hash from Cryptosystem (4 points)

Happy decided to build a hash function $H(M)$ out of the AES-128 encryption function E_k. First define $f(s, m) = E_m(s) \oplus m$, where s and m each have length 128. Let M be a message of arbitrary length. Here's how to compute $H(M)$.

- Pad M appropriately and divide it into 128-bit blocks $m_1 m_2 \dots m_t$.
- Compute the sequence:

$$\begin{aligned} s_1 &= m_1 \\ s_2 &= f(s_1, m_2) \\ s_3 &= f(s_2, m_3) \\ &\vdots \\ s_t &= f(s_{t-1}, m_t). \end{aligned}$$

- Define $H(M) = s_t$.

Questions

1. Given any $k \geq 1$ and 128-bit string s_k, show how to find a message $M = m_1 m_2 \dots m_k$ such that $H(M) = s_k$.

   [Hint: Use the fact that the decryption function $D_k()$ is the inverse of $E_k()$. This allows you to "work backwards" from s_k to s_1.]

2. Describe how to find a colliding pair (M, M') for $H()$.