ENEE 459-C, 9/21/15
Hashing Big Messages to 512 bits (or the Merkle-Damgard transform)

Assume you have a collision-resistant hash function h: ({0,1}^512,{0,1}^512) → {0,1}^{512} taking two inputs of 512 bits and outputting 512 bits.

QUESTION:

How do you hash a message m of L bits, where L is not necessarily a multiple of 512?

ANSWER:

Suppose L = k * 512 + x. Represent m as k+2 blocks m_1,m_2,...,m_{k+2} of 512 bits each. Block (k+1) is padded with 512-x zeros. Block (k+2) encodes the message length L in 512 bits.

To compute the hash d of the message m, you apply the chained transformation (known as Merkle-Damgard) as follows:

- Compute h_i=h(h_{i-1},m_i), for i=1,...,k+2. Note that h_0 is defined is the IV, which is public and hard-coded.
- The final output (i.e., the hash of the message d) is h_{k+2}.

QUESTION:

Why do we need to include the length of the message in the hash?

ANSWER:

Assume we do not. Consider now a message m of L bits. Again, L=k*512+x. We only create block k+1 but not block k+2. Now the hash is h_{k+1}.

Consider one can find an m' such that h(IV,m')=m', i.e., a fixed point for h. Then note, that the messages

m' m_1,m_2,...,m_{k+1}

and

m_1,m_2,...,m_{k+1}

have the same digest h^{k+1}, which is a collision! This is because h(IV,m')=m'. However, if we had encoder the length as block k+2, they would never have the same hash since the messages have different lengths.

QUESTION:

Is it easy to find fixed points for collision-resistant hash functions? Can you find one for the function h(x,m)=x XOR Enc_x(m)? This compression function has been used in practice!

ANSWER:

To find a fixed point for the above we need to find an m' such that h(IV,m) = IV, for a fixed (publicly known) IV.

This implies that IV XOR Enc_IV(m') = IV => ENC_IV(m')=000...0 => m' = DEC_{IV}(000...0)