

# Solutions to the homework exercises

## Homework 7

### Question 2

The security definition is pretty much the same as in the fixed length case: The adversary is granted access to an authentication oracle, this time for arbitrary length messages. She is then asked to produce a forgery, i.e. a message-tag-pair that has not been previously output by the oracle. Note that the message length of the forgery is allowed to be different from any of the messages sent to the oracle.

The attack works as follows. First, the adversary sends any message  $m = m_1 \| m_2 \| \dots \| m_l$  to the oracle, with  $|m_1| = n$ . This results in a tag  $t$ . Now the adversary outputs  $(m \| m', t)$ , with  $|m'| = l$ ,  $m'_1 = m_1 \oplus t$ , and  $m'_i = m_i$  for all other  $i = 2, \dots, l$ . This attack works because  $m_1 = m_1 \oplus t_0 = t \oplus m'_1$ .

## Homework 8

### Question 2

Well if you have a collision, i.e. strings  $x$  and  $x'$  such that  $H'(x) = H'(x')$ , then you have a collision for  $H$ . A correct solution for this problem should be a formal reduction i.e. it starts by assuming you have an adversary  $\mathcal{A}$  that finds a collision for  $H'$  and then describing what the adversary  $\mathcal{A}'$  against  $H$  does that uses  $\mathcal{A}$  as a subroutine.

### Question 3

The adversary can just query some message  $m$  to get a tag  $t$ , and then output  $(m', t \oplus H(m) \oplus H(m'))$ .

### Question 4

Here we can use the result from Question 2 for the function  $f$  that, on input  $m = m_1 m_2 \| m_3 \dots$  with  $|m_i| = n$  outputs  $m_3$  (if the input is long enough, otherwise it outputs  $0^n$ ). That way a successful adversary is the following. Send  $m$  to the oracle to get a tag  $t$  and outputs  $(m', t')$  where  $m'$  and  $t'$  are obtained from  $m$  and  $t$  by flipping the first bit.