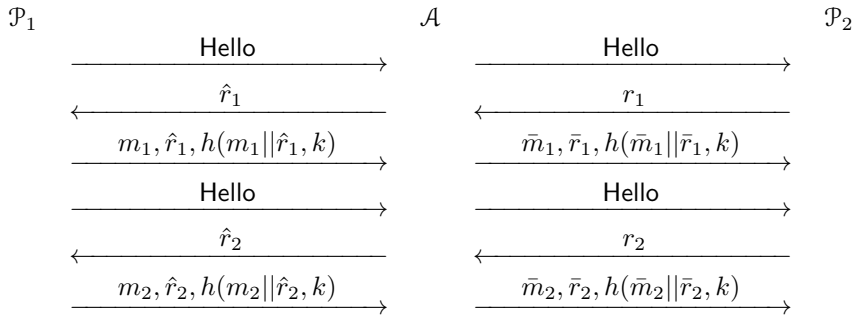
MTAT.07.003 Cryptology II
Spring 2012 / Exercise session ?? / Example Solution

Exercise (Stateless message authentication protocol). *Although a good message authentication code $h : \mathcal{M} \times \mathcal{K} \to \mathcal{T}$ protects against impersonation and substitution attacks, it does not guarantee security against reflection and interleaving attacks. In the following, we analyse the security of a simple three move protocol between $\mathcal{P}_1$ and $\mathcal{P}_2$. First, $\mathcal{P}_1$ sends* Hello *message to $\mathcal{P}_2$ who replies a randomly generated nonce $r \leftarrow \mathcal{R}$. To transfer a message m, $\mathcal{P}_1$ sends out $m, r, h(m||r, k)$ to $\mathcal{P}_2$ who accepts m only if the message authentication code is correct and the nonce r is the one sent out by $\mathcal{P}_2$. Let us consider security of two sequential invocations of this protocol between $\mathcal{P}_1$ and $\mathcal{P}_2$ in the setting where all messages are routed through the adversary $\mathcal{A}$.*



Since the adversary $\mathcal{A}$ is free to send messages at any time, there are many possible interleavings of protocol runs $\mathcal{A}$ can enforce. Analyse the security of the protocol provided that $m_1 \neq m_2$ are fixed and the attack $\mathcal{A}$ is considered successful if $\mathcal{P}_2$ accepts m_2, m_1 instead of m_1, m_2.

Solution. Although there is large number of potential interleavings, only some of them are really relevant. For instance, there is no difference in which order the first Hello messages are sent. Still, the order of some messages is really relevant. For instance, there is a big difference whether $\mathcal{A}$ knows r_1 before he or she sends out $\hat{r}_1$. For clarity, we first consider only pure attack strategies that always use a fixed interleaving patterns. Later we show how we can describe the success of mixed strategies in terms of pure attack strategies.

Let $a \prec b$ denote that the message a occurs before the message b in the protocol. Then a valid interleaving must satisfy certain restrictions by the construction, such as $r_1 \prec r_2$ and $\hat{r}_1 \prec \hat{r}_2$. Although these restrictions reduce the analysis space, the number of interleavings is still too large for exhaustive analysis. Therefore we further compact the analysis space further by showing that certain attack types are inferior compared to the other attack types. After that, we analyse only more successful attacks.

Simplification step. For further analysis, note that messages $r_1, \hat{r}_1, r_2$ and $\hat{r}_2$ are special in the protocol as these nonces are used to enforce causal relations between protocol messages that secure the protocol against simple reflection attacks. Hence, it makes sense to consider all possible interleavings of $r_1, \hat{r}_1, r_2$ and $\hat{r}_2$.

Lemma 1. *Let $\mathsf{Adv}(\mathcal{A})$ denote the success probability of an attack. Then for any attack $\mathcal{A}$ where $\hat{r}_1 \prec r_1$, there exists an attack $\mathcal{A}_*$ where $r_1 \prec \hat{r}_1$ such that $\mathsf{Adv}(\mathcal{A}) \leq \mathsf{Adv}(\mathcal{A}_*)$ and for any attack $\mathcal{A}$ where $\hat{r}_2 \prec r_2$, there exists an attack $\mathcal{A}_*$ where $r_2 \prec \hat{r}_2$ such that $\mathsf{Adv}(\mathcal{A}) \leq \mathsf{Adv}(\mathcal{A}_*)$.*

Proof. Let there be an attack $\mathcal{A}$ on the protocol where $\hat{r}_1 \prec r_1$. Then we can construct a new algorithm $\mathcal{A}_*$ that interacts with $\mathcal{A}$ in order to attack the protocol. Initially, $\mathcal{A}_*$ acts as a wire between $\mathcal{A}$ and a protocol, that is, it sends all messages acquired form the protocol to $\mathcal{A}$ and submit all the messages from $\mathcal{A}$ as corresponding protocol messages. This type of execution can continue until $\mathcal{A}$ sends out $\hat{r}_1$ to $\mathcal{P}_1$. Then $\mathcal{A}_*$ initiates communication with $\mathcal{P}_2$, sending him Hello and receives r_1 from $\mathcal{P}_2$. Next, $\mathcal{A}_*$ stores the value r_1 and continues by sending out $\hat{r}_1$ to $\mathcal{P}_1$. After that $\mathcal{A}_*$ again acts as a wire between the protocol and the $\mathcal{A}$ with a small exception. When $\mathcal{A}$ queries r_1, $\mathcal{A}_*$ returns the stored value of r_1. This modification guarantees that $r_1 \prec \hat{r}_1$ in the modified attack. Also, it is evident that $\mathcal{P}_2$ reaches the same state as in the original protocol, since all received and sent messages are identical in both runs. Thus, $\mathsf{Adv}(\mathcal{A}) = \mathsf{Adv}(\mathcal{A}_*)$. The second claim can be proven by similarly delaying $\hat{r}_2$. □

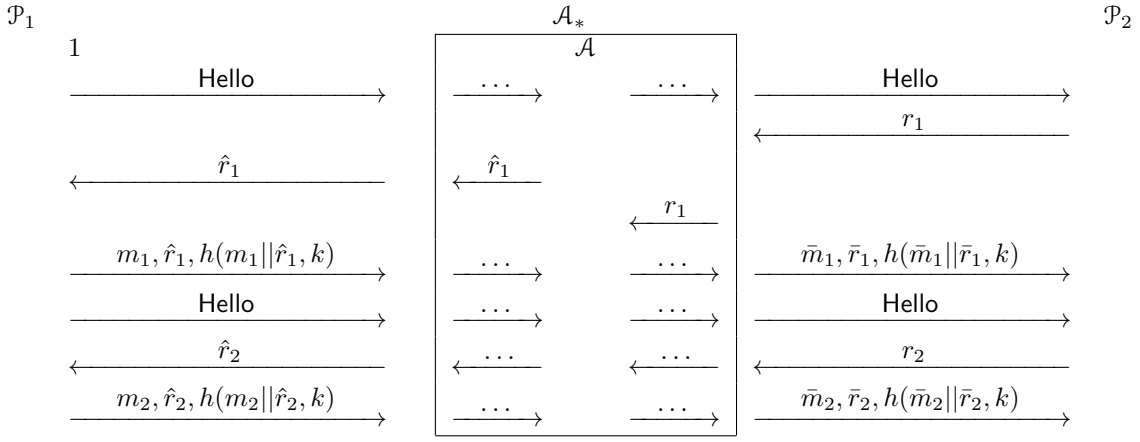
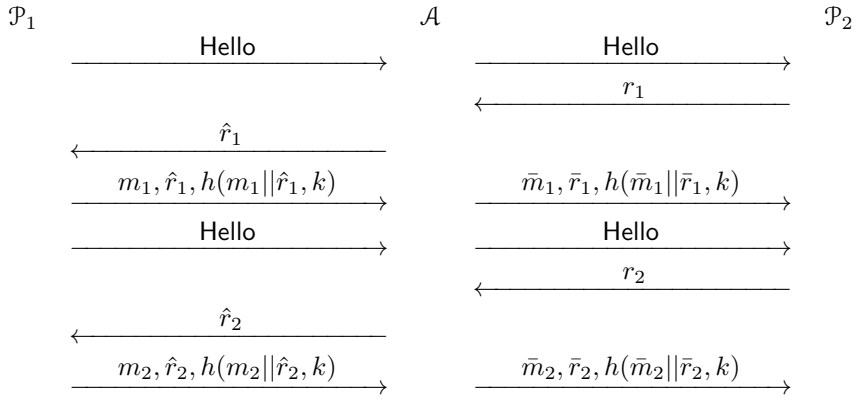


Figure 1: Schematic illustration of the delay construction $\mathcal{A}_*$.

Analysis of a particular scheduling. As a next step towards the full proof, let us analyse the attack where $\mathcal{A}$ when $r_1 \prec \hat{r}_1 \prec r_2 \prec \hat{r}_2$. The potential interleaving pattern is depicted below.



The attack is successful if $\mathcal{P}_2$ accepts m_2, m_1 instead of m_1, m_2. Note that $\mathcal{P}_2$ accepts both messages only if $r_i = \bar{r}_i$ and $h(\bar{m}_i||r_i, k) = \bar{t}_i$ for $i \in \{1, 2\}$. Let us analyse the first check more closely. In this case, the adversary $\mathcal{A}$ sees a message authentication code $h(m_1||\hat{r}_1, k)$ before $\mathcal{A}$ must provide a valid tag $\bar{t}_1$. However, the tag t_1 is not directly useful, since $\bar{m}_1 = m_2$ by our requirements. In fact, a successful attack can be viewed as a substitution attack, where $\mathcal{A}$ replaces the message $m_1, \hat{r}_1$ with m_1, r_1.

Thus, we can limit the success probability by direct reduction to the security of the message authentication code. Recall that the security of a message authentication code is defined through the following game

$\mathcal{Q}^{\mathcal{B}}$
$k \xleftarrow{u} \mathcal{K}$
$t_0 \leftarrow \perp$
For $i \in \{1, \ldots, q\}$ do
  $x_i \leftarrow \mathcal{B}(t_{i-1})$
  $t_i \leftarrow h(x_i, k)$
$(x, t) \leftarrow \mathcal{B}(t_q)$
return $[t \stackrel{?}{=} h(x, k)] \wedge [(x, t) \notin \{(x_1, t_1), \ldots, (x_q, t_q)\}]$.

As $\mathcal{A}$ receives only one message authentication tag, we can consider the simplified security game

$$
\begin{array}{l}
\mathcal{Q}^{\mathcal{B}} \\
\left[\begin{array}{l}
k \xleftarrow{u} K \\
x_1 \leftarrow \mathcal{B} \\
t_1 \leftarrow h(x_1, k) \\
(x, t) \leftarrow \mathcal{B}(x_1, t_1) \\
\textbf{return } x_1 \neq x \wedge h(x, k) = t \ ,
\end{array}\right.
\end{array}
$$

which defines $(t, 1, \varepsilon)$-secure message authentication codes. Hence, we can construct $\mathcal{B}$ as follows

$$
\begin{array}{l}
\mathcal{B} \\
\left[\begin{array}{l}
r_1 \leftarrow \mathcal{R} \\
\hat{r}_1 \leftarrow \mathcal{A}(r_1) \\
x_1 \leftarrow m_1 || \hat{r}_1 \\
\textbf{return } x_1
\end{array}\right.
\end{array}
\qquad\qquad
\begin{array}{l}
\mathcal{B}(t_1) \\
\left[\begin{array}{l}
\bar{t}_1 \leftarrow \mathcal{A}(m_1, \hat{r}_1, t_1) \\
x = m_2 || r_1 \\
\textbf{return } (x, \bar{t}_1)
\end{array}\right.
\end{array}
$$

if we omit all Hello messages received and sent by $\mathcal{A}$ and resolve ambiguities caused by the ordering of messages similarly to the analysis carried out in Lemma 1. By substituting $\mathcal{B}$ into $\mathcal{Q}$, we get a game

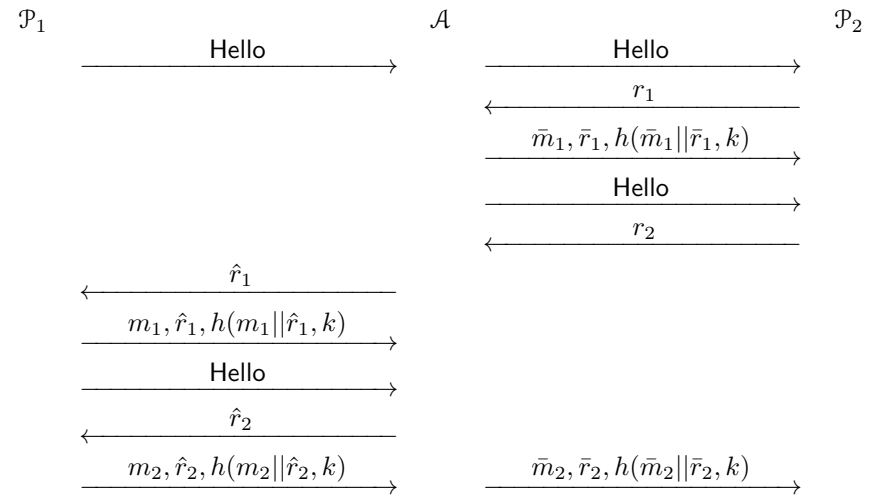
$$
\begin{array}{l}
\mathcal{G}_0 \\
\left[\begin{array}{l}
k \xleftarrow{u} K \\
r_1 \leftarrow \mathcal{R} \\
\hat{r}_1 \leftarrow \mathcal{A}(r_1) \\
\bar{t}_1 \leftarrow \mathcal{A}(m_1, \hat{r}_1, h(m_1 || \hat{r}_1, k)) \\
\textbf{return } [m_2 || r_1 \neq m_1 || \hat{r}_1] \wedge [h(m_2 || r_1, k) \stackrel{?}{=} \bar{t}_1]
\end{array}\right.
\end{array}
$$

that ends with one only if $\mathcal{P}_2$ accepts the first message. Thus we have proved

$$
\Pr[\mathcal{A} \text{ succeeds}] \leq \Pr[\mathcal{P}_2 \text{ accepts } m_2 \text{ as the first message}] \leq \mathsf{Adv}_h^{\mathsf{mac}}(\mathcal{B}) \ .
$$

Since the running time of $\mathcal{B}$ is only by a constant larger than the running-time, we can bound $\mathsf{Adv}_h^{\mathsf{mac}}(\mathcal{B}) \leq \varepsilon$.

Analysis of a particular scheduling. Let us analyse the attack where $\mathcal{A}$ when $r_1 \prec r_2 \prec \hat{r}_1 \prec \hat{r}_2$. The potential interleaving pattern is depicted below.



In this case, $\mathcal{A}$ obtains no message authentication tags before $\mathcal{A}$ has to construct $\bar{t}_1$, i.e., $\mathcal{A}$ conducts an impersonation attack. Consequently, $(t, 0, \varepsilon)$-secure message authentication codes are sufficient to bound the success probability. Since the previously analysed attack pattern required $(t, 1, \varepsilon)$-secure message authentication codes, it makes sense to construct the reduction to this more liberal security game. One of the possible reductions is given below

$$\begin{array}{ll} \mathcal{C} & \mathcal{C}(t_1) \\ \left\lceil \begin{array}{l} r_* \leftarrow \mathcal{R} \\ x_1 \leftarrow m_1 || r_* \\ \textbf{return } x_1 \end{array} \right. & \left\lceil \begin{array}{l} r_1 \leftarrow \mathcal{R} \\ \bar{m}_1, \bar{r}_1, \bar{t}_1 \leftarrow \mathcal{A}(r_1) \\ x = m_2 || r_1 \\ \textbf{return } (x, \bar{t}_1) \ . \end{array} \right. \end{array}$$

A direct substitution of $\mathcal{C}$ into $\mathcal{Q}$ leads to

$$\mathcal{G}_1 \left\lceil \begin{array}{l} k \xleftarrow{u} K \\ r_* \leftarrow \mathcal{R} \\ t_1 \leftarrow h(m_1 || r_*, k) \\ r_1 \leftarrow \mathcal{R} \\ \bar{m}_1, \bar{r}_1, \bar{t}_1 \leftarrow \mathcal{A}(r_1) \\ \textbf{return } [m_1 || r_* \neq m_2 || r_1] \wedge h(m_2 || r_1, k) = \bar{t}_1 \ . \end{array} \right.$$

By deleting all irrelevant computations, we get

$$\mathcal{G}_1 \left\lceil \begin{array}{l} k \xleftarrow{u} K \\ r_1 \leftarrow \mathcal{R} \\ \bar{m}_1, \bar{r}_1, \bar{t}_1 \leftarrow \mathcal{A}(r_1) \\ \textbf{return } h(m_2 || r_1, k) = \bar{t}_1 \ . \end{array} \right.$$

Note that the last line is corresponds to a sub-check that must hold in order for $\mathcal{P}_2$ to accept the message. Hence, we have again proven

$$\Pr[\mathcal{A} \text{ succeeds}] \leq \Pr[\mathcal{P}_2 \text{ accepts } m_2 \text{ as the first message}] \leq \mathsf{Adv}_h^{\mathsf{mac}}(\mathcal{C}) \ .$$

Analysis of mixed attack patterns. To get a final security bound, we must first bound the the success for all adversaries with the fixed interleaving patterns. Although there are $4! = 24$ possible combinations of $r_1, r_2, \hat{r}_1, \hat{r}_2$, not all of them are possible. Clearly, attacks where $r_2 \prec r_1$ or $\hat{r}_2 \prec \hat{r}_1$ are impossible. From the remaining patterns there are three where $\bar{r}_1 \prec r_1$. Lemma 1 assured that these attacks can be converted to the attacks $r_1 \prec \bar{r}_1$ without being less successful, so we can omit these. From the remaining three attack patterns there is one where $\bar{r}_2 \prec r_2$. We can omit it as Lemma 1 guarantees that there is an attack with $r_2 \prec \bar{r}_2$ where the adversary is as successful. As a consequence, we are left with two attack patterns: $r_1 \prec \bar{r}_1 \prec r_2 \prec \hat{r}_2$ and $r_1 \prec r_2 \prec \hat{r}_1 \prec \hat{r}_2$, whose success we have previously analysed.

If we apply the same analysis for the adversaries that adaptively choose the interleaving pattern during the attack, then we can conclude that without of loss of generality we can consider adversaries which dynamically decide between the attacks $r_1 \prec \bar{r}_1 \prec r_2 \prec \hat{r}_2$ and $r_1 \prec r_2 \prec \hat{r}_1 \prec \hat{r}_2$.

Given such an adversary $\mathcal{A}$, we can construct two adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$ such that $\mathcal{A}_1$ follows the interleaving pattern $r_1 \prec \bar{r}_1 \prec r_2 \prec \hat{r}_2$ and $\mathcal{A}_2$ follows the interleaving pattern $r_1 \prec r_2 \prec \hat{r}_1 \prec \hat{r}_2$. The corresponding reduction is trivial, we run $\mathcal{A}$ and halt if it chooses the other attack strategy. Obviously, $\mathcal{A}$ succeeds only if either $\mathcal{A}_1$ or $\mathcal{A}_2$ succeeds. Since $\mathcal{A}_1$ and $\mathcal{A}_2$ use fixed attack strategies, we can conclude that for any t-time adversary $\mathcal{A}$:

$$\Pr[\mathcal{A} \text{ succeeds}] \leq \Pr[\mathcal{A}_1 \text{ succeeds}] + \Pr[\mathcal{A}_2 \text{ succeeds}] \leq 2\varepsilon$$

if h is a $(t, 1, \varepsilon)$-secure message authentication code. This reduction is not optimal, as we could directly reduce the success of $\mathcal{A}$ to MAC forgeries. However, the corresponding reduction is technically more complicated.