1 current sec model to transcribe ADEM + AMAC to a more symmetric style

1.1 used primitives

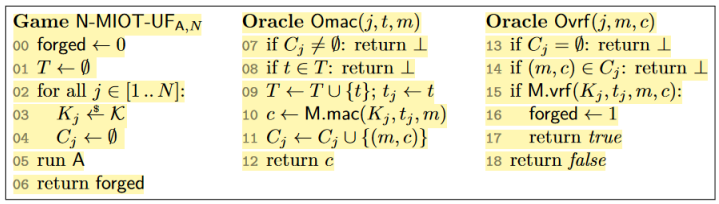
- ADEM: input tag, key and message lead to a cythertext. It should be improbable distinguish the cythertexts of two messages. (adversary may choose two cythertexts and has to guess which one of the two is encrypted)
- AMAC: input tag, key and message lead to a cythertext. It should be improbable to make a forgery (a pair (key, tag, message, cythertext) that verifies without begin generated by calling Omac(key, tag, message) first)

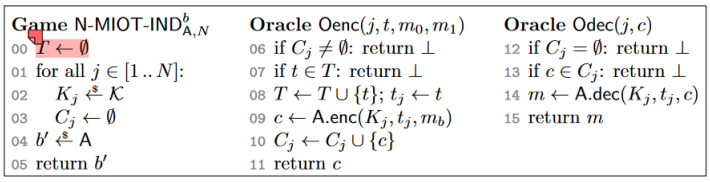
1.2 goal

Each user is provided with two keys, a message and a tag that is bound to the user and does not repeat between users. The message is encrypted using the tag and two keys to generate a cythertext consisting of two parts. First part is Cdem which is the message encrypted with the nonce and the first key while the second part is Cmac that is the mac computed over Cdem, the tag and the second key. Given only one queries to Oenc per user and multiple queries to Odec which always occurs after the Oenc queries, the message should be protected against active adversaries as long as ADEM and AMAC are secure.

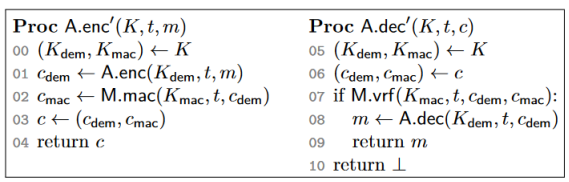
1.3 Sec model

the security is purely based on the games for the AMAC and ADEM that are visible below.





with



2 needed sec model to transcribe ADEM + AMAC to a more symmetric style

for now we only look at the nonce based options as the pkc paper does that too.

2.1 used primitives

- ADEM: input nonce, key en message lead to a cythertext which should be improbable to distinguish from RO (adversary has to guess if he is talking to RO or ADEM)
- AMAC: input nonce, key en message lead to a tag that should be improbable to distinguishable from random oracle (adversary has to guess if he is talking to RO or AMAC)

2.2 goal

Each user is provided with two keys, a message and a lock that does not repeat between users. The message is encrypted using the lock and two keys. Given only one queries to Oenc per user and multiple queries to Odec which always occurs after the Oenc queries, the message should be protected against active adversaries as long as ADEM and AMAC are secure.

2.3 Sec model

We define the following sec games for the AMAC, the ADEM and the ADEM+AMAC (names will prob be improved later):

header	**header**
0 : $t \leftarrow g$	0 : $t \leftarrow g$

Game $\text{AMAC}^b_{A,N}$	**Oracle** Omac(j,n,m)
$used_n \leftarrow \emptyset$	if $\text{T}_j \neq \emptyset$: return $\perp$
for j $\in [1..N]$:	if n $\in used_n$: return $\perp$
$\quad \text{K}_j \leftarrow^{\$} K$	$used_n \leftarrow used_n \cup \{n\}$
$\quad \text{T}_j \leftarrow \emptyset$	$\text{n}_j \leftarrow n$
b' $\leftarrow^{\$} A$	if b=0: t $\leftarrow$ M.mac(K_j,n_j,m)
return b'	if b=1: t $\leftarrow$ RO.mac(K_j,n_j,m)
	$\text{T}_j \leftarrow \text{T}_j \cup \{t\}$
	return t

Table 1: AMAC game

Game $\text{ADEM}^b_{A,N}$	**Oracle** Oenc(j,n,m)
$used_n \leftarrow \emptyset$	if $\text{C}_j \neq \emptyset$: return $\perp$
for j $\in [1..N]$:	if n $\in used_n$: return $\perp$
$\quad \text{K}_j \leftarrow^{\$} K$	$used_n \leftarrow used_n \cup \{n\}$
$\quad \text{T}_j \leftarrow \emptyset$	if b=0: c $\leftarrow$ E.enc(K_j,n,m)
b' $\leftarrow^{\$} A$	if b=1: c $\leftarrow$ RO.enc(K_j,n,m)
return b'	$\text{C}_j \leftarrow \text{C}_j \cup \{c\}$
	return c

Table 2: ADEM game

Game $\text{ADEM}^b_{A,N}$	**Oracle** Oenc(j,n,m)	**Oracle** Odec(j,n,c)
$used_n \leftarrow \emptyset$	if $\text{C}_j \neq \emptyset$: return $\perp$	if (c,n) $\in \text{C}_j$: return $\perp$
for j $\in [1..N]$:	if n $\in used_n$: return $\perp$	if b = 0:
$\text{K}_j \leftarrow^{\$} K$	$used_n \leftarrow used_n \cup \{n\}$	m $\leftarrow$A.dec'(K_j,n,c)
$\text{T}_j \leftarrow \emptyset$	$\text{n}_j \leftarrow n$	return m
b' $\leftarrow^{\$} A$	if b=0:	if b = 1:
return b'	c $\leftarrow$ E.enc'(K_j,n_j,m)	m $\leftarrow$RO.dec'(K_j,n,c)
	if b=1:	return m
	c $\leftarrow$ RO.enc'(K_j,n_j,m)	
	$\text{C}_j \leftarrow \text{C}_j \cup \{(c,n)\}$	
	return c	

Table 3: ADEM + AMAC game

where E.enc, E.dec, M.mac , RO.enc, RO.dec and RO.mac are inherited from the underlying primitives and E.enc', E.dec', RO.enc' and RO.dec' will be defined later here defined.

3 burning questions

-

4 current todo's

- sec model aanpassen naar context
- kijken of we message of dist based games willen (zijn deze twee ubehaud fundementeel anders?)
- de games opnieuw opschrijven
- de games aanpassen naar context
- crypto.bib kijken

5 main idea

The PKC paper ends with a ADEM + AMAC construction as "solution". The original paper from ENC -¿ MAC has been revised, so this should prob be revised as well. In general its nice to write down thing in a more "sym crypto" style as we use symmetric primitives. It would probably also be nice to revise it more in general and see what other ways there are to reach the end-goal expected in the PKC paper.