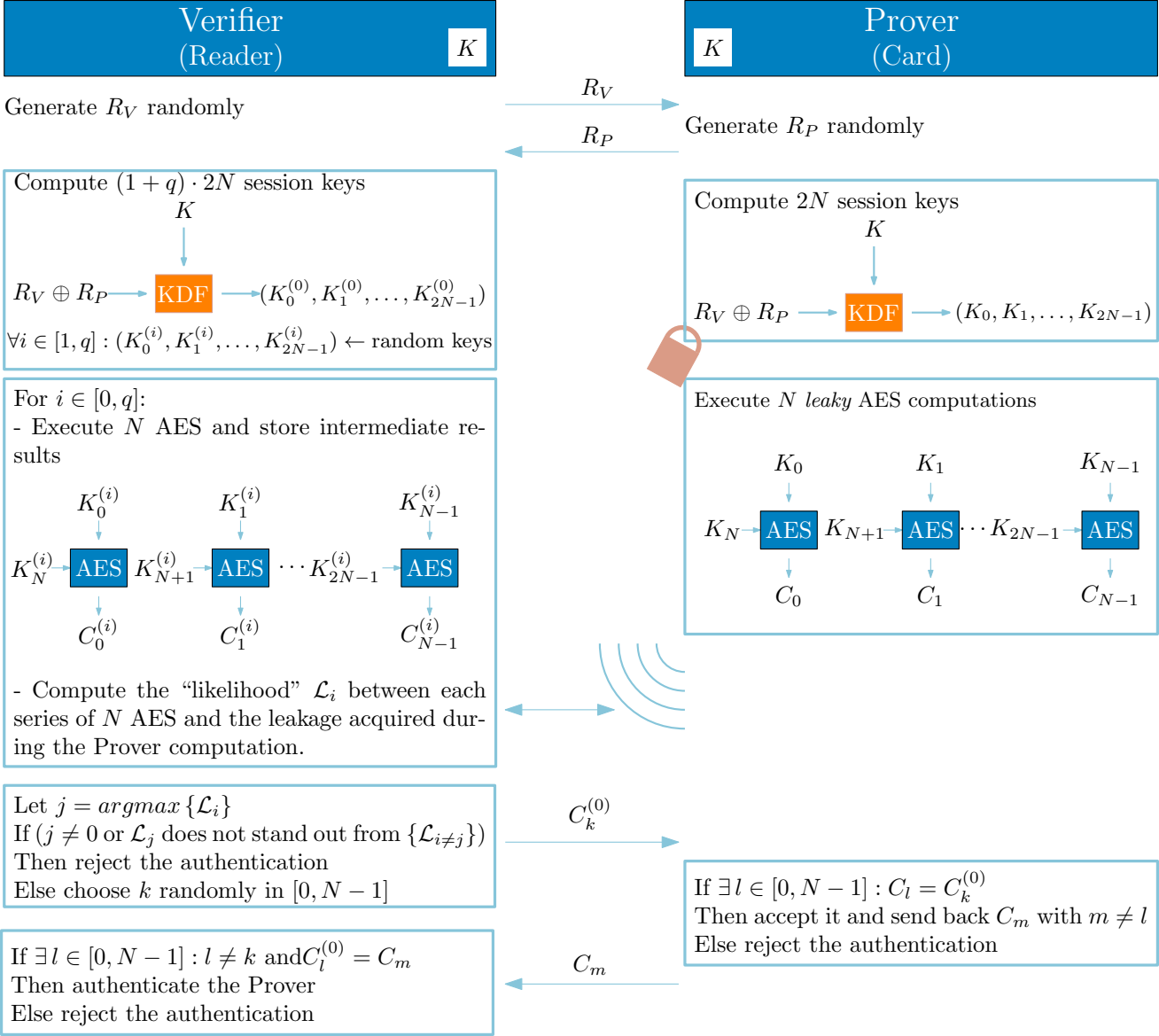
Verifier
(Reader)
K
Prover
(Card)
K
Generate R_V randomly
R_V
R_P
Generate R_P randomly
Compute $(1+q)\cdot 2N$ session keys
K
$R_V \oplus R_P$
KDF
$(K_0^{(0)}, K_1^{(0)}, \ldots, K_{2N-1}^{(0)})$
$\forall i \in [1,q] : (K_0^{(i)}, K_1^{(i)}, \ldots, K_{2N-1}^{(i)}) \leftarrow$ random keys
Compute $2N$ session keys
K
$R_V \oplus R_P$
KDF
$(K_0, K_1, \ldots, K_{2N-1})$
For $i \in [0,q]$:
- Execute N AES and store intermediate results
$K_0^{(i)}$ $K_1^{(i)}$ $K_{N-1}^{(i)}$
$K_N^{(i)}$ AES $K_{N+1}^{(i)}$ AES $\cdots K_{2N-1}^{(i)}$ AES
$C_0^{(i)}$ $C_1^{(i)}$ $C_{N-1}^{(i)}$
- Compute the "likelihood" $\mathcal{L}_i$ between each series of N AES and the leakage acquired during the Prover computation.
Execute N *leaky* AES computations
K_0 K_1 K_{N-1}
K_N AES K_{N+1} AES $\cdots K_{2N-1}$ AES
C_0 C_1 C_{N-1}
Let $j = argmax\{\mathcal{L}_i\}$
If ($j \neq 0$ or $\mathcal{L}_j$ does not stand out from $\{\mathcal{L}_{i\neq j}\}$)
Then reject the authentication
Else choose k randomly in $[0, N-1]$
$C_k^{(0)}$
If $\exists l \in [0,N-1] : C_l = C_k^{(0)}$
Then accept it and send back C_m with $m \neq l$
Else reject the authentication
If $\exists l \in [0,N-1] : l \neq k$ and$C_l^{(0)} = C_m$
Then authenticate the Prover
Else reject the authentication
C_m