Institute of Computer Science, University of Bern
Cryptography, HS 2019

Prof. Christian Cachin
Luca Zanolini

Exercise 2

2.1 Basics on libraries (5pt)

1. Below are two calling programs $\mathcal{A}_1, \mathcal{A}_2$ and two libraries $\mathcal{L}_1, \mathcal{L}_2$ with a common interface:

$\mathcal{A}_1$

$r_1 := \text{RAND}(6)$
$r_2 := \text{RAND}(6)$
return $r_1 \stackrel{?}{=} r_2$

$\mathcal{A}_2$

$r := \text{RAND}(6)$
return $r \stackrel{?}{\geqslant} 3$

$\mathcal{L}_1$

$\text{RAND}(n)$:
$r \leftarrow \mathbb{Z}_n$
return r

$\mathcal{L}_2$

$\text{RAND}(n)$:
return 0

- What is $\Pr[\mathcal{A}_1 \diamond \mathcal{L}_1 \Rightarrow 1]$?
- What is $\Pr[\mathcal{A}_1 \diamond \mathcal{L}_2 \Rightarrow 1]$?
- What is $\Pr[\mathcal{A}_2 \diamond \mathcal{L}_1 \Rightarrow 1]$?
- What is $\Pr[\mathcal{A}_2 \diamond \mathcal{L}_2 \Rightarrow 1]$?

2. In each problem, a pair of libraries are described. State whether or not $\mathcal{L}_{\text{left}} \equiv \mathcal{L}_{\text{right}}$. If so, show how they assign identical probabilities to all outcomes. If not, then describe a successful *distinguisher*.

   Assume that both libraries use the same value of n. Does your answer ever depend on the choice of n?

   Note that $\bar{x}$ denotes the bitwise-complement of x and $x \,\&\, y$ denotes the bitwise AND of the two strings:

- **$\mathcal{L}_{\text{left}}$**

  $\text{QUERY}()$:
  $x \leftarrow \{0,1\}^n$
  return x

  $\mathcal{L}_{\text{right}}$

  $\text{QUERY}()$:
  $x \leftarrow \{0,1\}^n$
  $y := \bar{x}$
  return y

- **$\mathcal{L}_{\text{left}}$**

  $\text{QUERY}()$:
  $x \leftarrow \mathbb{Z}_n$
  return x

  $\mathcal{L}_{\text{right}}$

  $\text{QUERY}()$:
  $x \leftarrow \mathbb{Z}_n$
  $y := 2x \,\%\, n$
  return y

- **$\mathcal{L}_{\text{left}}$**

  $\text{QUERY}()$:
  $x \leftarrow \{0,1\}^n$
  $y \leftarrow \{0,1\}^n$
  return $x \,\&\, y$

  $\mathcal{L}_{\text{right}}$

  $\text{QUERY}()$:
  $z \leftarrow \{0,1\}^n$
  return z

2.2 Security of a modified One-time Pad (OTP) (3pt)

Suppose we modify one-time pad to add a few $\texttt{0}$ bits to the end of every ciphertext:

$\mathcal{K} = \{\texttt{0},\texttt{1}\}^\lambda$ $\mathcal{M} = \{\texttt{0},\texttt{1}\}^\lambda$ $\mathcal{C} = \{\texttt{0},\texttt{1}\}^{\lambda+2}$	KeyGen: $k \leftarrow \{\texttt{0},\texttt{1}\}^\lambda$ return k	Enc(k, m): $c := k \oplus m$ return $c\|\|\texttt{00}$	Dec(k, c): remove last 2 bits of c $m := k \oplus c$ return m

(In Enc, $||$ refers to concatenation of strings.) Show that the resulting scheme still satisfies one-time secrecy. Your proof can use the fact that one-time pad has one-time secrecy.

2.3 Construction of a distinguisher (2pt)

Show that the following encryption scheme does not have one-time secrecy, by constructing a program that distinguishes the two relevant libraries from the one-time secrecy definition.

$\mathcal{K} = \{1, \cdots, 9\}$ $\mathcal{M} = \{1, \cdots, 9\}$ $\mathcal{C} = \mathbb{Z}_{10}$	KeyGen: $k \leftarrow \{1, \cdots, 9\}$ return k	Enc(k, m): return $k \times m \ \% \ 10$

2.4* Size of the OTP key space (Bonus: +3pt)

Prove that if an encryption scheme Σ has $|\Sigma.\mathcal{K}| < |\Sigma.\mathcal{M}|$ then it cannot satisfy one-time secrecy. Try to structure your proof as an explicit attack on such a scheme (i.e., a distinguisher against the appropriate libraries).

You may consider Enc to be a deterministic function, as in the one-time pad. To obtain even more bonus points, prove this statement for randomized Enc. However, you may assume that Dec is deterministic.

Hint: The definition of interchangeabilitiy doesn't care about the running time of the distinguisher or the calling program. So even an exhaustive brute-force attack would be valid.