# Stream ciphers: definition and IV

Tanja Lange

Eindhoven University of Technology

2WF80: Introduction to Cryptology

# Reminder: one-time pad

- Let $m \in \{0,1\}^\ell$, i.e., a message is a string of $\ell$ bits.
  Let $k \in \{0,1\}^\ell$, chosen uniformly at random.
  Then $c = m + k$, where addition is done modulo 2 in each position.
  (In more mathematical notation: $m, k \in \mathbb{F}_2^\ell, c = m + k$.)

```
  011001110011001010010010111001
+ 010111101100011010110100101001
--------------------------------
  001110011111010000100110010000
```

- The one-time pad is information-theoretically secure –
  there is no information about the plaintext in the ciphertext.
  $c_i = 0$ can come from $m_i = k_i = 0$ or from $m_i = k_i = 1$.
  $c_i = 1$ can come from $m_i = 0, k_i = 1$ or from $m_i = 1, k_i = 0$.

# Reminder: one-time pad

- Let $m \in \{0,1\}^\ell$, i.e., a message is a string of $\ell$ bits.
  Let $k \in \{0,1\}^\ell$, chosen uniformly at random.
  Then $c = m + k$, where addition is done modulo 2 in each position.
  (In more mathematical notation: $m, k \in \mathbb{F}_2^\ell, c = m + k$.)

```
  0110011100110010100100101110 01
+ 010111101100011010110100101001
--------------------------------
  001110011111010000100110010000
```

- The one-time pad is information-theoretically secure –
  there is no information about the plaintext in the ciphertext.
  $c_i = 0$ can come from $m_i = k_i = 0$ or from $m_i = k_i = 1$.
  $c_i = 1$ can come from $m_i = 0, k_i = 1$ or from $m_i = 1, k_i = 0$.
- This requires the key to be as long as the message –
  the "two-time" pad is insecure.
  This makes the scheme unusable for most situations.

# Stream ciphers

- Alice and Bob share a “short” key (typically 128 - 256 bits).
- Stream ciphers take a key as input and generate long stream of pseudorandom numbers (typically bits or bytes).
- A good stream cipher produces a stream of numbers that
  - is unpredictable given any previous portion of the stream;
  - does not exhibit any non-random statistical properties.
- A minimum requirement is that the cipher output passes a battery of statistical tests, such as the Diehard tests.
- Encryption with a stream cipher works the same as with the OTP:

$$c = m + s,$$

  where $s$ is the stream cipher output.

# Two-time pad

If Tom uses the same pad twice and if his messages always start with `DEAR ALICE,` (using ASCII encoding in hexadecimal, addition mod 16)

```
m1 =  D  E  A  R     A  L  I  C  E  ,     L  E  T  '  S     M  E  E  T
m1 = 44 45 41 52 20 41 4C 49 43 45 2C 20 4C 45 54 27 53 20 4D 45 45 54
 p = 54 48 49 53 20 49 53 20 54 4F 54 41 4C 4C 59 20 52 41 4E 44 4F 4D
c1 = 98 8D 8A A5 40 8A 9F 69 97 84 70 61 88 81 AD 47 A5 61 8B 89 84 91

m2 = 44 45 41 52 20 41 4C 49 43 45 2C 20 54 4F 44 41 59 20 49 20 43 41
 p = 54 48 49 53 20 49 53 20 54 4F 54 41 4C 4C 59 20 52 41 4E 44 4F 4D
c2 = 98 8D 8A A5 40 8A 9F 69 97 84 70 61 90 8B 9D 61 AB 61 87 64 82 8E
```

then Eve notices the common start of the messages. This tells her

1. Tom is reusing `p`,
2. the messages start with the same text.

(Anything else would be too much of a coincidence).

If Eve can get `m1`, e.g. by observing that the message went to Alice and then observing Alice meet Tom, she gets $\mathtt{p}=\mathtt{c1}-\mathtt{m1}$ and $\mathtt{m2}=\mathtt{c2}-\mathtt{p}$.

In any case, subtracting the ciphertexts gives $\mathtt{m1}-\mathtt{m2}$ (no sign of `p`).

This issue is not specific to OTP, same for stream ciphers.

# Stream ciphers

- Encryption with a stream cipher works the same as with the OTP:

$$c = m + s,$$

  where $s$ is the stream cipher output.
- We must avoid the issues of the two-time pad.
  Given the description so far this means
  - Alice and Bob must remember how many output numbers they have used and continue from that point on.
  - Next message requires recomputing all past steps or keeping a state.
  - Lost messages desynchronize Alice and Bob.

# Stream ciphers

- Encryption with a stream cipher works the same as with the OTP:

$$c = m + s,$$

  where $s$ is the stream cipher output.
- We must avoid the issues of the two-time pad.
  Given the description so far this means
  - Alice and Bob must remember how many output numbers they have used and continue from that point on.
  - Next message requires recomputing all past steps or keeping a state.
  - Lost messages desynchronize Alice and Bob.
- Solve these issues by including an Initialization Vector (IV) so that

$$S : \{0,1\}^v \times \{0,1\}^\ell \to \{0,1\}^*, \quad (IV, k) \mapsto s.$$

  Typically, the output length is limited to some $n$, (a bound on) the length of the message to be encrypted.
- Encryption computes

$$(IV, m, s) \mapsto c = (IV, m + s).$$