Stream Cipher

Santanu Sarkar

IIT Madras, Chennai

Stream Cipher Principle: Basic Idea

Parties: A (Sender/Receiver) and B (Receiver/Sender)

Procedure

- A and B share a stream of random data (keystream) Z_i, where $i = 0, 1, \ldots$

Stream Cipher Principle: Basic Idea

Parties: A (Sender/Receiver) and B (Receiver/Sender)

Procedure

- A and B share a stream of random data (keystream) Z_i, where $i = 0, 1, \ldots$
- The plaintext stream M_i is XOR-ed with Z_i to generate the cipher stream C_i. $[C_i = M_i \oplus Z_i]$

Stream Cipher Principle: Basic Idea

Parties: A (Sender/Receiver) and B (Receiver/Sender)

Procedure

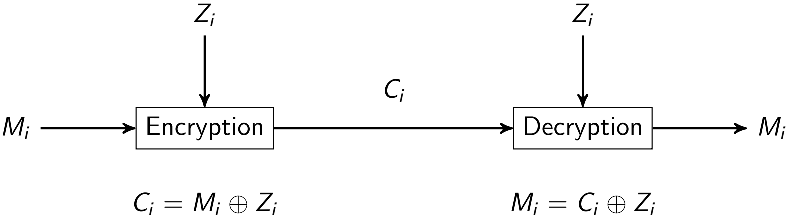
- A and B share a stream of random data (keystream) Z_i, where $i = 0, 1, \ldots$
- The plaintext stream M_i is XOR-ed with Z_i to generate the cipher stream C_i. $[C_i = M_i \oplus Z_i]$
- The cipher stream C_i is XOR-ed with Z_i to generate the plaintext stream M_i. $[M_i = C_i \oplus Z_i]$

Stream Cipher Principle: Basic Idea

Parties: A (Sender/Receiver) and B (Receiver/Sender)

Procedure

- A and B share a stream of random data (keystream) Z_i, where $i = 0, 1, \ldots$
- The plaintext stream M_i is XOR-ed with Z_i to generate the cipher stream C_i. $[C_i = M_i \oplus Z_i]$
- The cipher stream C_i is XOR-ed with Z_i to generate the plaintext stream M_i. $[M_i = C_i \oplus Z_i]$



Pseudorandom Generator

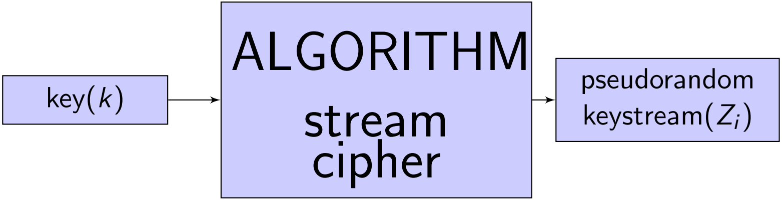
- The security of the message depends on the randomness of Z.

Pseudorandom Generator

- The security of the message depends on the randomness of Z.
- A and B share a small binary string called *key*

Pseudorandom Generator

- The security of the message depends on the randomness of Z.
- A and B share a small binary string called *key*
- After that, the algorithm will take *key* as input and keep on generating *random-looking bitstream*, the keystream bits. This algorithm is the Stream cipher.



RC4

RC4

- RC4 was designed by Ron Rivest in 1987.
- Made public in 1994.

RC4

- RC4 was designed by Ron Rivest in 1987.
- Made public in 1994.
- Twenty years from its release, RC4 was the most used stream cipher in the world.

RC4

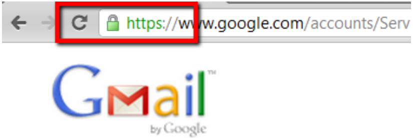
- RC4 was designed by Ron Rivest in 1987.
- Made public in 1994.
- Twenty years from its release, RC4 was the most used stream cipher in the world.
- Used in WEP(Wired Equivalent Privacy) in 1997 and later in WPA(Wi-fi Protected Access).

RC4

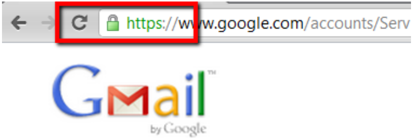
- RC4 was designed by Ron Rivest in 1987.
- Made public in 1994.
- Twenty years from its release, RC4 was the most used stream cipher in the world.
- Used in WEP(Wired Equivalent Privacy) in 1997 and later in WPA(Wi-fi Protected Access).
- Used by Google and Microsoft.

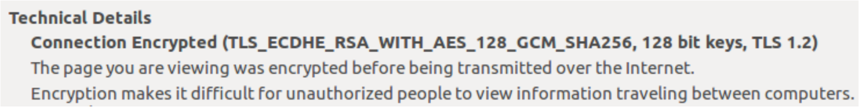
RC4

- RC4 was designed by Ron Rivest in 1987.
- Made public in 1994.
- Twenty years from its release, RC4 was the most used stream cipher in the world.
- Used in WEP(Wired Equivalent Privacy) in 1997 and later in WPA(Wi-fi Protected Access).
- Used by Google and Microsoft.



https://www.google.com/accounts/Serv





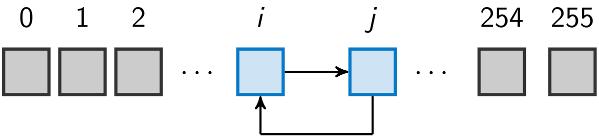
Technical Details

Connection Encrypted (TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, 128 bit keys, TLS 1.2)

The page you are viewing was encrypted before being transmitted over the Internet.

Encryption makes it difficult for unauthorized people to view information traveling between computers.

Key Scheduling Algorithm (KSA)

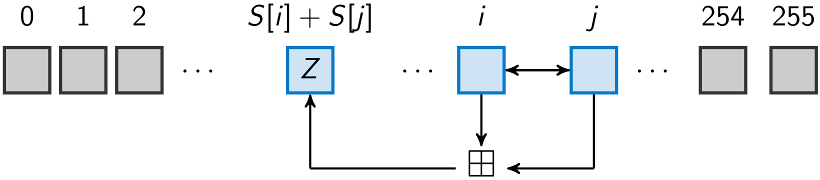


```
Initialize index: $j = 0$;
for $i = 0, \ldots, 255$ do
    $j = j + S[i] + K[i]$;
    Swap $S[i] \leftrightarrow S[j]$;
end
```

INPUT: *S*-array initialized to identity permutation, and key K

OUTPUT: Scrambled *S*-array

Pseudo-Random Generation Algorithm (PRGA)



```
Initialize indices: $i = j = 0$;
while $TRUE$ do
    $i = i + 1$;
    $j = j + S[i]$;
    Swap $S[i] \leftrightarrow S[j]$;
    Output $Z = S[S[i] + S[j]]$;
end
```

INPUT: Scrambled S-array, obtained as the KSA output

OUTPUT: Pseudo-random stream

An ideal stream cipher should generate all the numbers from 0 to 255 with equal probabilities, i.e, $\frac{1}{256}$.

An ideal stream cipher should generate all the numbers from 0 to 255 with equal probabilities, i.e, $\frac{1}{256}$.

Biases of RC4:

An ideal stream cipher should generate all the numbers from 0 to 255 with equal probabilities, i.e, $\frac{1}{256}$.

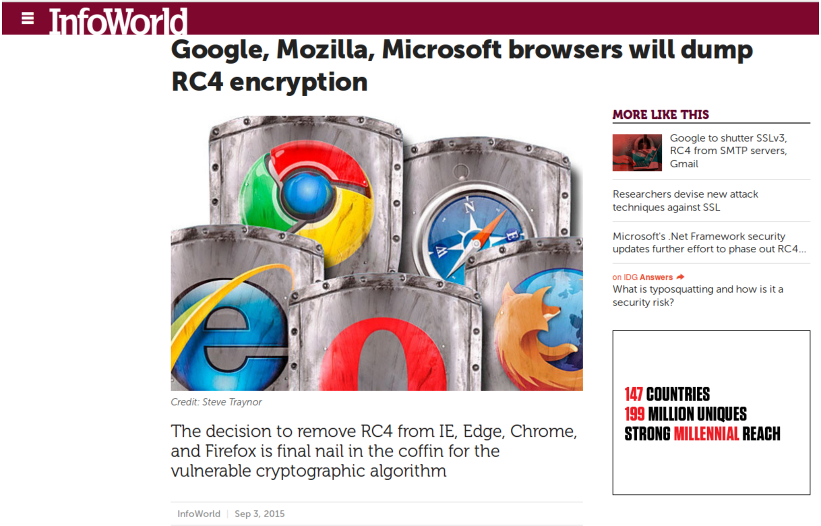
Biases of RC4:

- $\Pr(Z_2 = 0) = \frac{2}{256}$ Mantin et al in 2001.
- $\Pr(Z_r = r) > \frac{1}{256}$ Isobe et al in 2013.

An ideal stream cipher should generate all the numbers from 0 to 255 with equal probabilities, i.e, $\frac{1}{256}$.

Biases of RC4:

- $\Pr(Z_2 = 0) = \frac{2}{256}$ Mantin et al in 2001.
- $\Pr(Z_r = r) > \frac{1}{256}$ Isobe et al in 2013.
- $\Pr(Z_r = r - K[0]) < \frac{1}{256}$: Paterson et al in 2014.



≡ InfoWorld

Google, Mozilla, Microsoft browsers will dump RC4 encryption

Credit: Steve Traynor

The decision to remove RC4 from IE, Edge, Chrome, and Firefox is final nail in the coffin for the vulnerable cryptographic algorithm

InfoWorld | Sep 3, 2015

MORE LIKE THIS

- Google to shutter SSLv3, RC4 from SMTP servers, Gmail
- Researchers devise new attack techniques against SSL
- Microsoft's .Net Framework security updates further effort to phase out RC4...

on IDG Answers ➦
What is typosquatting and how is it a security risk?

Thank You