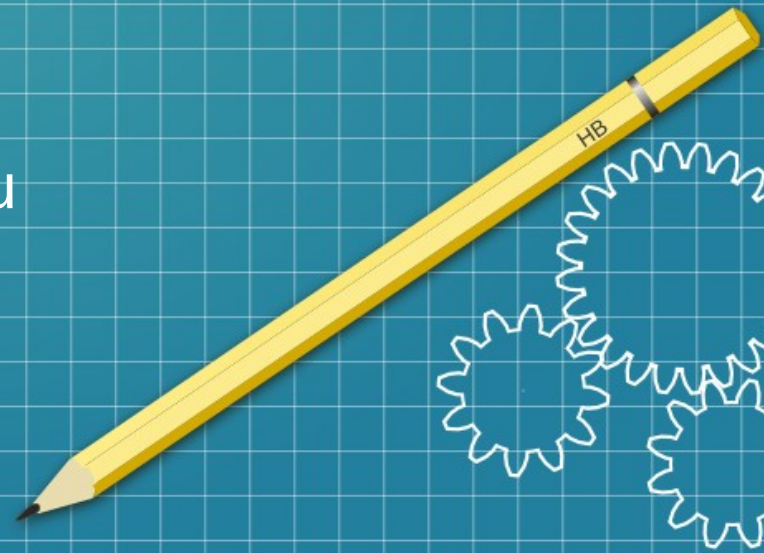


# Stream ciphers

CSE 468 Fall 2024  
jedimaestro@asu.edu



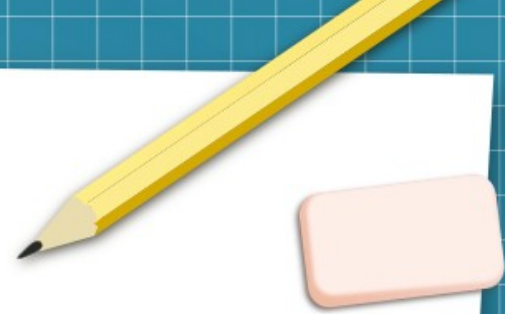


# KRACK attacks...

<https://www.youtube.com/watch?v=fZ1R9RliM1W>

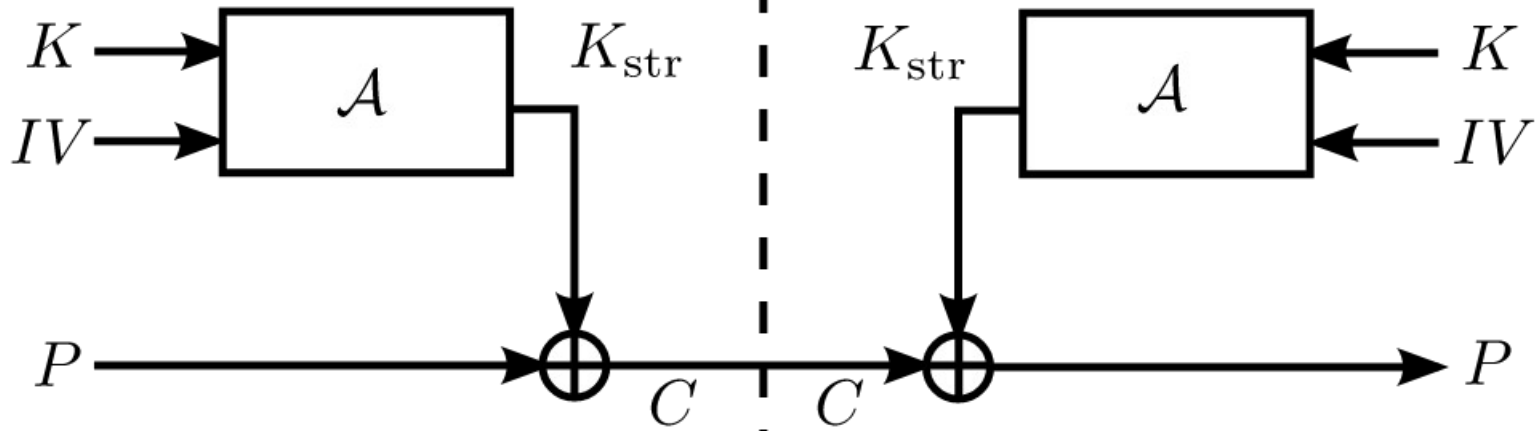
# Outline

- Why stream ciphers?
- WEP
  - IVs reused because of birthday principle
- WPA2
  - IVs reused because of key re-installation (KRACK attacks)
- ShadowSocks
  - Redirection attack due to malleability
- Other examples



# Encryption

# Decryption

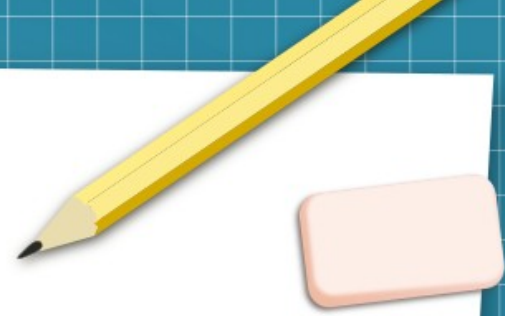


# Good things about stream ciphers



- Can pre-compute key material, encryption/decryption is just XOR
- Can send small bursts without wasting space on padding
- More modular implementation in hardware
  - IV and key are only inputs
- Some stream ciphers that are not based on block ciphers are very fast
  - *E.g.*, RC4

# Playing with fire?



- You should NEVER reuse key material
  - Harder than it sounds
    - Handshake protocols, *etc.* might have replay attacks
    - APIs, education
    - Downgrade attacks
- You should NEVER assume that successful decryption is the same as authentication
  - Even worse to assume this than it is for block ciphers

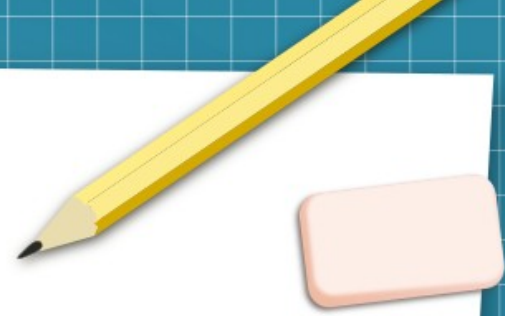
A yellow pencil and a pink eraser are positioned in the top right corner of the white paper, appearing to be part of the document's design.

A theme we saw in asymmetric cryptography...

Crypto protocols and network protocols  
sometimes don't play nicely together.  
(Messages can be lost, modified, replayed,  
dropped, *etc.*)

# WEP

- IV is only 24 bits
- No real authentication
  - CRC is not a cryptographic hash function





# WEP encryption

## “Wired Equivalent Privacy”

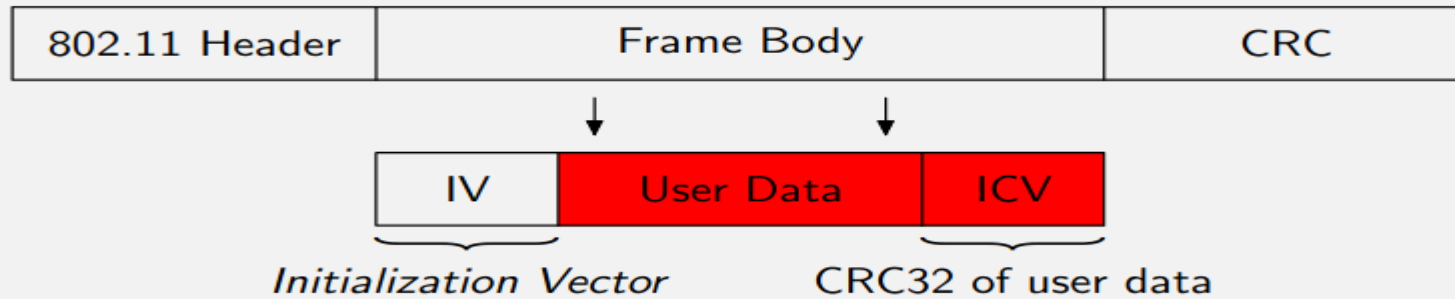
-Have to be physically in a building to plug in, have to know the passphrase to join WiFi (or do you?)

## RC4, 40-bit key, 24-bit IV

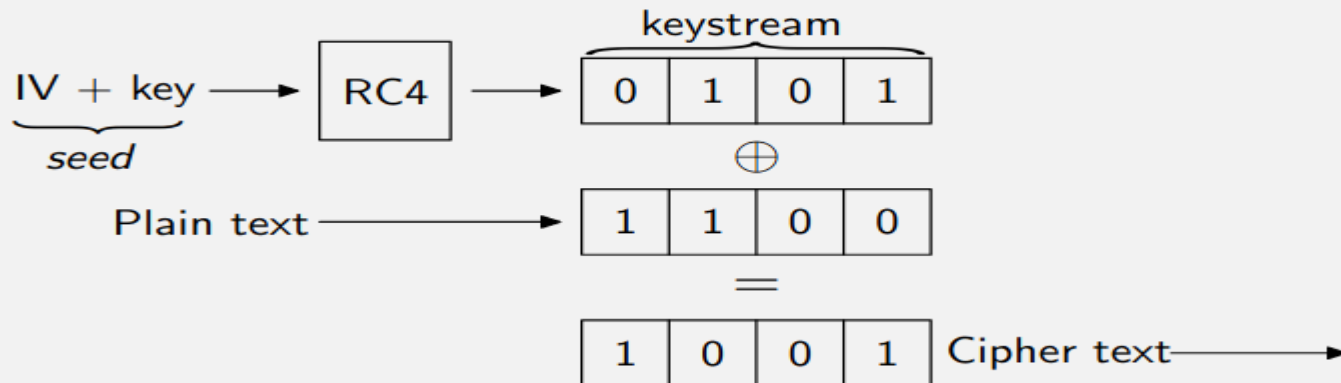
Following are from:

<https://jedcrandall.github.io/courses/cse468fall2022/wep/198fbe890b692e5296fcf7ad1b015e653ec9.pdf>

## Data frame format



## Encryption



If cipher-text & plain-text pair is known, their XOR is a keystream.  
Known plain-text (LLC/SNAP headers) in IP packets:

802.11 header	0xAA	0xAA	0x03	0x00	0x00	0x00	0x08	0x00
---------------	------	------	------	------	------	------	------	------

$\oplus$

802.11 header	Cipher-text							
---------------	-------------	--	--	--	--	--	--	--

=

8 bytes of keystream								
----------------------	--	--	--	--	--	--	--	--

Can recover 8 bytes of keystream by eavesdropping a packet.

- Can encrypt (and transmit) 8 bytes of arbitrary data.

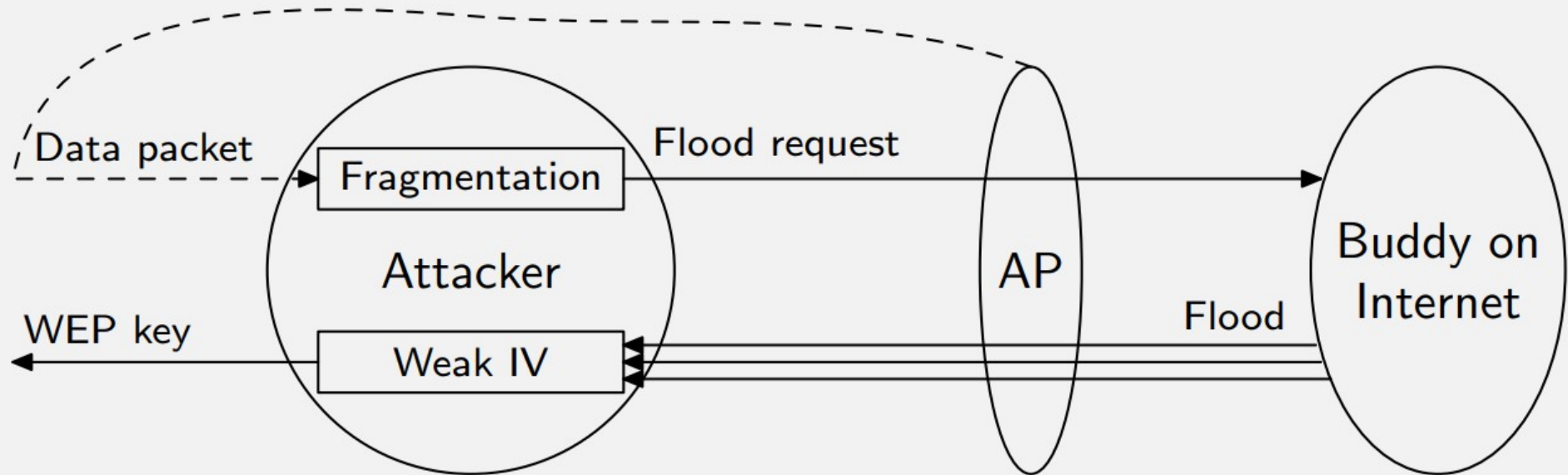
## Possible to create statistical biases in the Key Scheduling Algorithm (KSA)

### More info:

<https://www.youtube.com/watch?v=2o3Hs-JDWLs>

# Crack WEP key in minutes...

## Operation of wesside

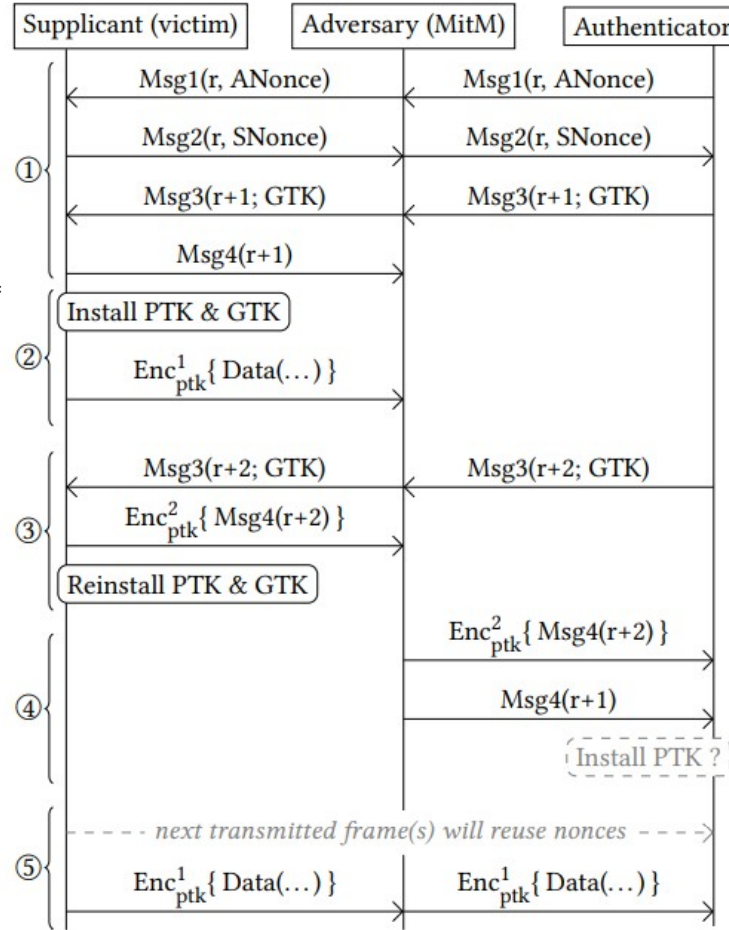


# WPA2

- IV is 48 bits (128-bit key with AES in a special counter mode called CCMP)
- SHA1 HMAC for authentication (called a MIC)
  - 160 bits



<https://papers.mathyvanhoef.com/ccs2017.pdf>



**KRACK**  
attacks

**Figure 4: Key reinstatement attack against the 4-way handshake, when the supplicant (victim) still accepts plaintext retransmissions of message 3 if a PTK is installed.**

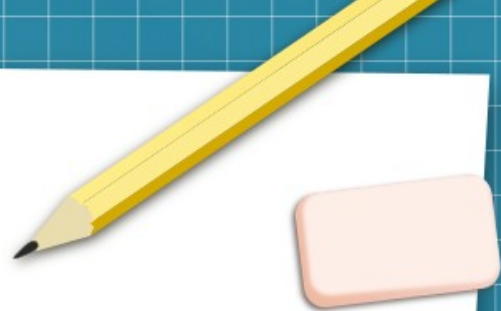
# ShadowSocks



- Let's the user choose between non-AEAD and AEAD ciphers, with many options for each
  - AEAD = Authenticated Encryption with Associated Data
  - Most implementations don't support AEAD
    - No authentication of messages

Following is from... <https://www.idcoffer.com/wp-content/uploads/2020/02/Redirect-attack-on-Shadowsocks-stream-ciphers.pdf>





## **Ciphers of shadowsocks:**

Shadowsocks support the two kinds of ciphers:

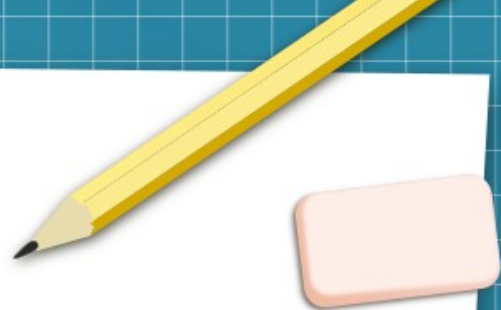
Stream ciphers (none-AEAD cipher):

Rc4-md5, salsa20, chacha20, chacha-ietf, aes-ctf, bf-cfb, camellia-cfb, aes-cfb

AEAD ciphers:

aes-gcm, chacha-ietf-poly1305, xchacha20-ietf-poly1305

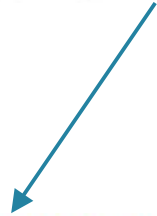
# What is ShadowSocks?



The Shadowsocks local component (`ss-local`) acts like a traditional SOCKS5 server and provides proxy service to clients. It encrypts and forwards data streams and packets from the client to the Shadowsocks remote component (`ss-remote`), which decrypts and forwards to the target. Replies from target are similarly encrypted and relayed by `ss-remote` back to `ss-local`, which decrypts and eventually returns to the original client.

**client <---> ss-local <--[encrypted]--> ss-remote <---> target**

**[target address][payload]**



Addresses used in Shadowsocks follow the SOCKS5 address format:

**[1-byte type][variable-length host][2-byte port]**

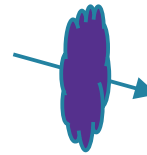
The following address types are defined:

0x01: host is a 4-byte IPv4 address.

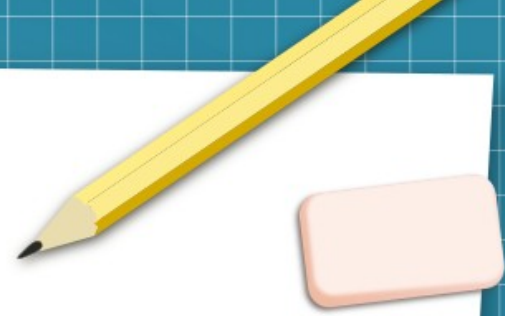
0x03: host is a variable length string, starting with a 1-byte length, followed by up to 255-byte domain name.

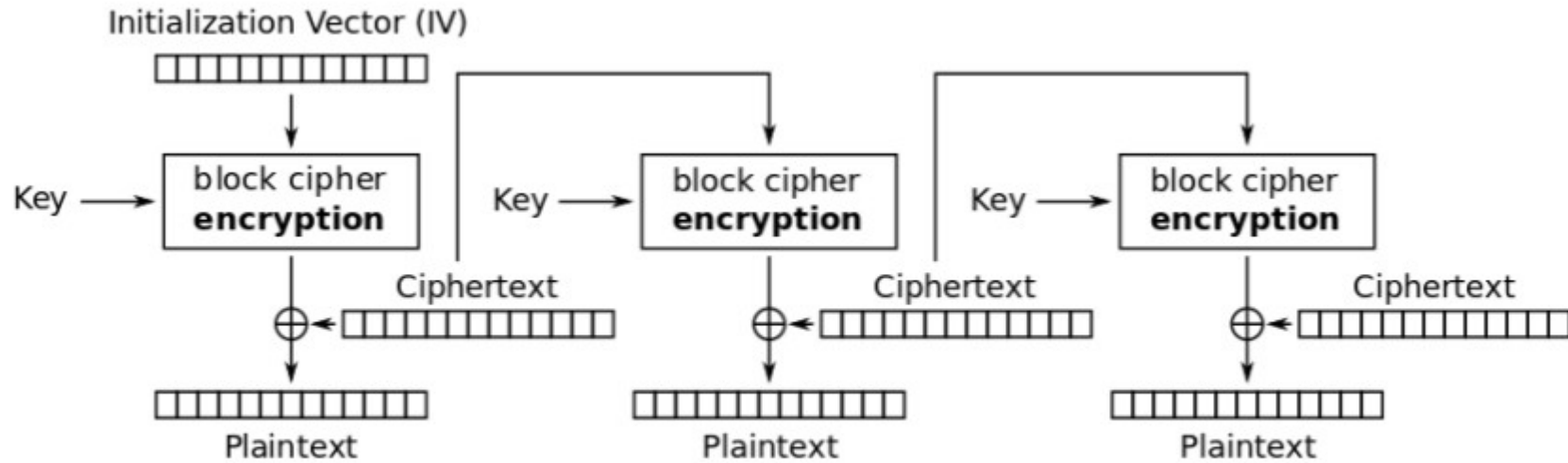
0x04: host is a 16-byte IPv6 address

The port number is a 2-byte big-endian unsigned integer.



**[IV][encrypted payload]**





Cipher Feedback (CFB) mode decryption

IVs are chosen randomly, transmitted in plaintext.

```
GET /html/en/reference/matrices/_sources/sage/mat
Host: doc.sagemath.org
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64)
Accept: text/html,application/xhtml+xml,application/javascript
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: __cfduid=ddc36b5813d7782ce467edb33058f732
__utma=138969649.1329315963.1545386824.1545394846
sphinxsidebar=visible; _gid=GA1.2.1229955866.1548
If-None-Match: W/"5c45d22a-127"
If-Modified-Since: Mon, 21 Jan 2019 14:07:38 GMT
```

```
HTTP/1.1 304 Not Modified
Date: Sat, 26 Jan 2019 09:59:47 GMT
Connection: keep-alive
Via: 1.1 varnish
Cache-Control: max-age=600
ETag: W/"5c45d22a-127"
Expires: Sat, 26 Jan 2019 10:09:47 GMT
Age: 0
```

```
root@DESKTOP-3UNO8NU:/mnt/g/code/shadowsocks/decrypt# nc -l -p 4626 >1.txt
```

```
^Z[10] Killed nc -l -p 4626 > 1.txt
```

```
[11]+ Stopped nc -l -p 4626 > 1.txt
```

```
root@DESKTOP-3UNO8NU:/mnt/g/code/shadowsocks/decrypt# cat 1.txt
```

```
1 304 Not Found Sat, 26 Jan 2019 07:15:21 GMT
```

```
Connection: close
```

```
Via: 1.1 varnish
```

```
Cache-Control: max-age=600
```

```
ETag: W/"5c45d22a-127"
```

```
Expires: Sat, 26 Jan 2019 06:59:41 GMT
```

```
Age: 0
```

```
X-Served-By: cache-pao17445-PAO
```

```
X-Cache: MISS
```

```
X-Cache-Hits: 0
```

```
X-Timer: S1548486922.795009, VS0, VE25
```

```
Vary: Accept-Encoding
```

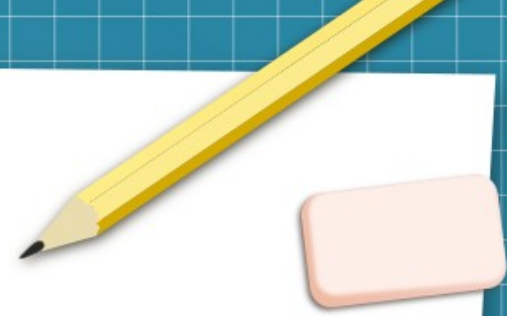
```
X-Fastly-Request-ID: 7f80e83d2fe5428bb3e38bb4e7d472af1b22eb4b
```

```
Server: cloudflare
```

```
CF-RAY: 49f1301d27589408-SJC
```



Many other stream cipher fails...

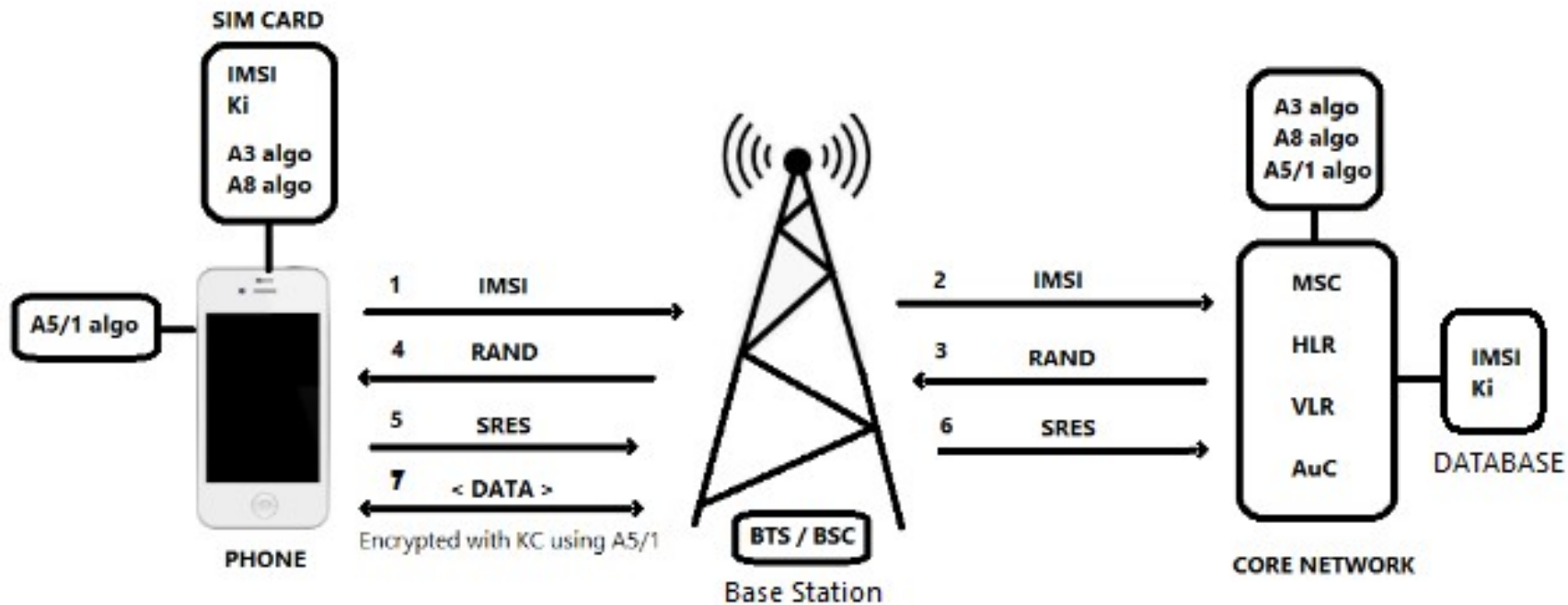


[https://en.wikipedia.org/wiki/Enigma\\_machine#/media/File:Enigma \(crittografia\) - Museo\\_scienza\\_e\\_tecnologia\\_Milano.jpg](https://en.wikipedia.org/wiki/Enigma_machine#/media/File:Enigma_(crittografia) - Museo_scienza_e_tecnologia_Milano.jpg)

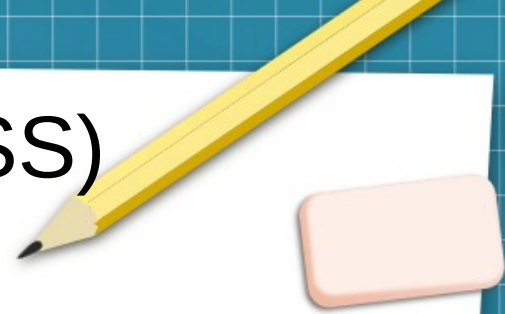




[https://en.wikipedia.org/wiki/Type\\_B\\_Cipher\\_Machine#/media/File:Photograph\\_of\\_RED\\_cryptographic\\_device\\_-\\_National\\_Cryptologic\\_Museum\\_-\\_DSC07863.JPG](https://en.wikipedia.org/wiki/Type_B_Cipher_Machine#/media/File:Photograph_of_RED_cryptographic_device_-_National_Cryptologic_Museum_-_DSC07863.JPG)



# Content Scramble System (CSS)



# High-bandwidth Digital Content Protection



[https://commons.wikimedia.org/wiki/File:Apple\\_TV,\\_1st\\_generation\\_-\\_mainboard\\_-\\_Silicon\\_Image\\_SiI1930CTU-3215.jpg](https://commons.wikimedia.org/wiki/File:Apple_TV,_1st_generation_-_mainboard_-_Silicon_Image_SiI1930CTU-3215.jpg)



This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License. It makes use of the works of Mateus Machado Luna.

