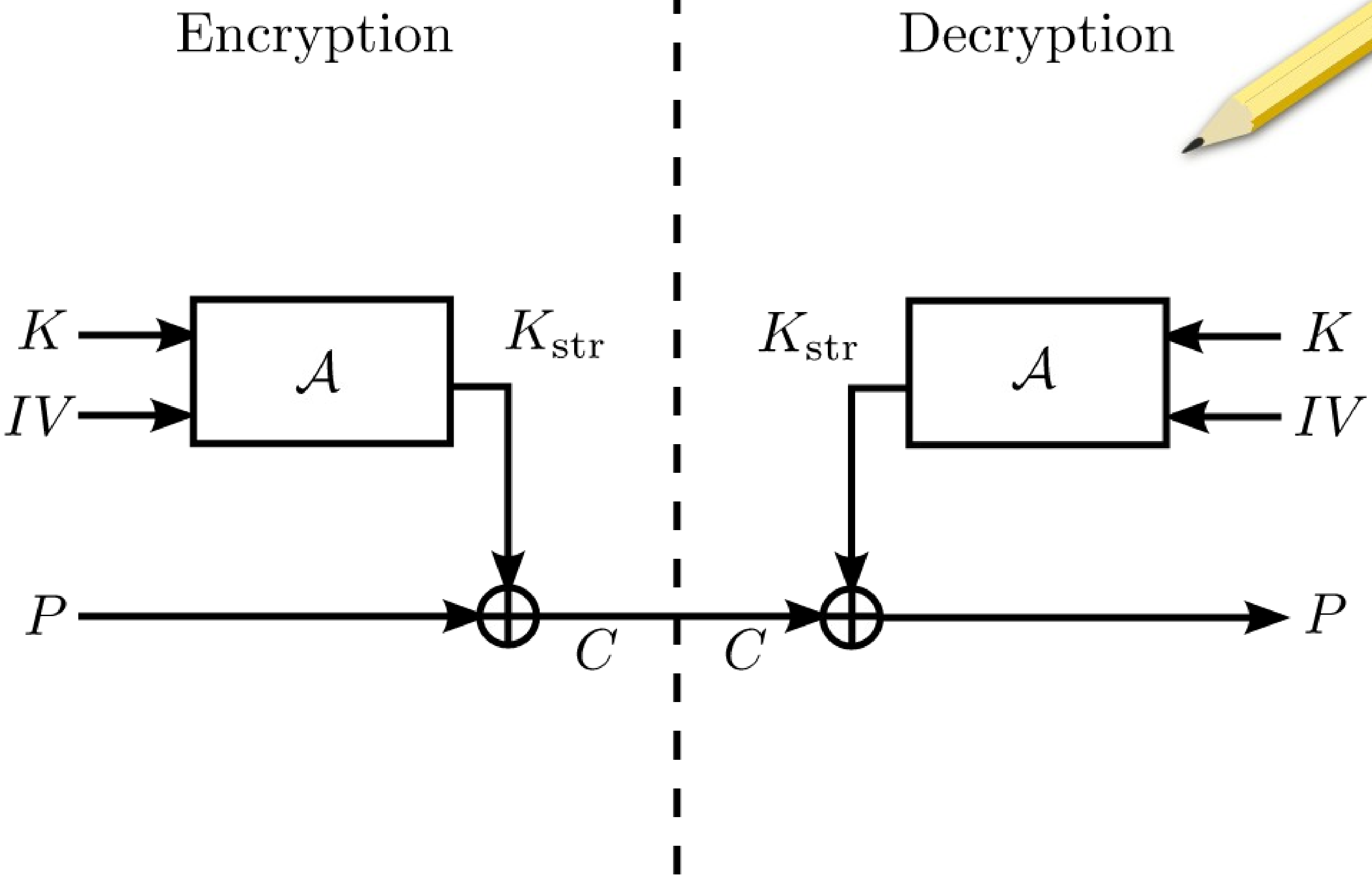
Stream ciphers and WiFi security

CSE 468 Fall 2025
jedimaestro@asu.edu

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



https://commons.wikimedia.org/wiki/File:Stream_cipher.svg

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 theme we will see in asymmetric cryptography…

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

WiFi security

Basically three use cases

- -Open
- -Personal (e.g., a passphrase)
- -Enterprise

https://securityuncorked.com/2022/07/wifi-security-the-3-types-of-wifi-networks/

WiFi security in a nutshell

WEP is very

- Can be broken in seconds/minutes

WPA was only a stop gap

- RC4 hardware

WPA2 is maybe okay for now if you do it right?

- Notion of personal *vs.* enterprise introduced here
- KRACK attacks

WPA3 is better, maybe?

- Dragonblood attacks
- Open no longer means just “unencrypted”

WEP

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

WEP encryption

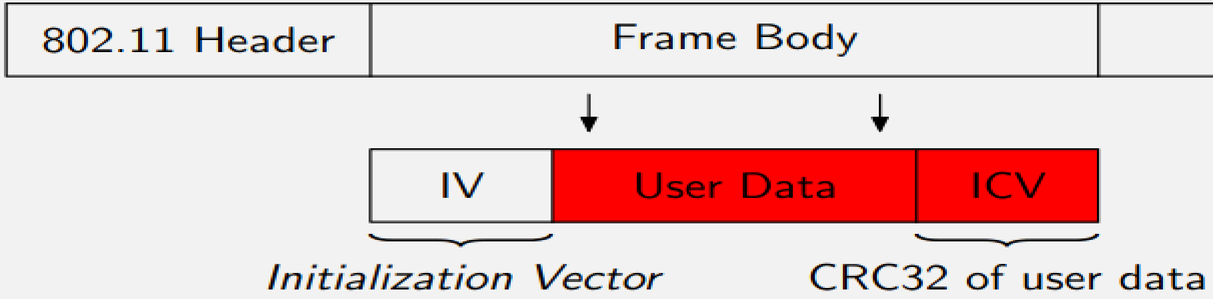
“Wired Equivalent Privacy”

-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



802.11 Header

Frame Body

CRC

IV

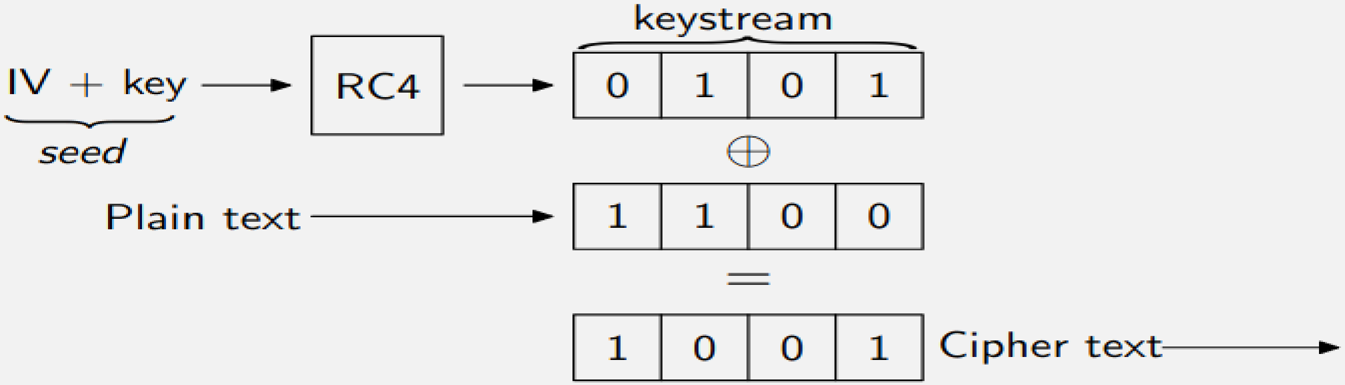
User Data

ICV

Initialization Vector

CRC32 of user data

Encryption



IV + key

seed

RC4

keystream

0	1	0	1

⊕

Plain text

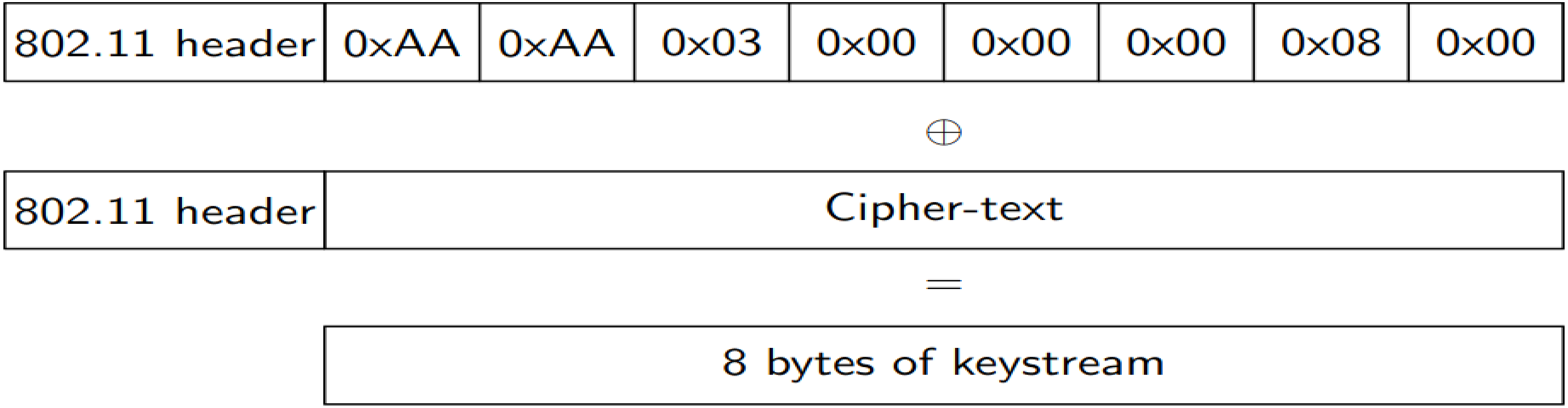
1	1	0	0

=

1	0	0	1

Cipher text

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.

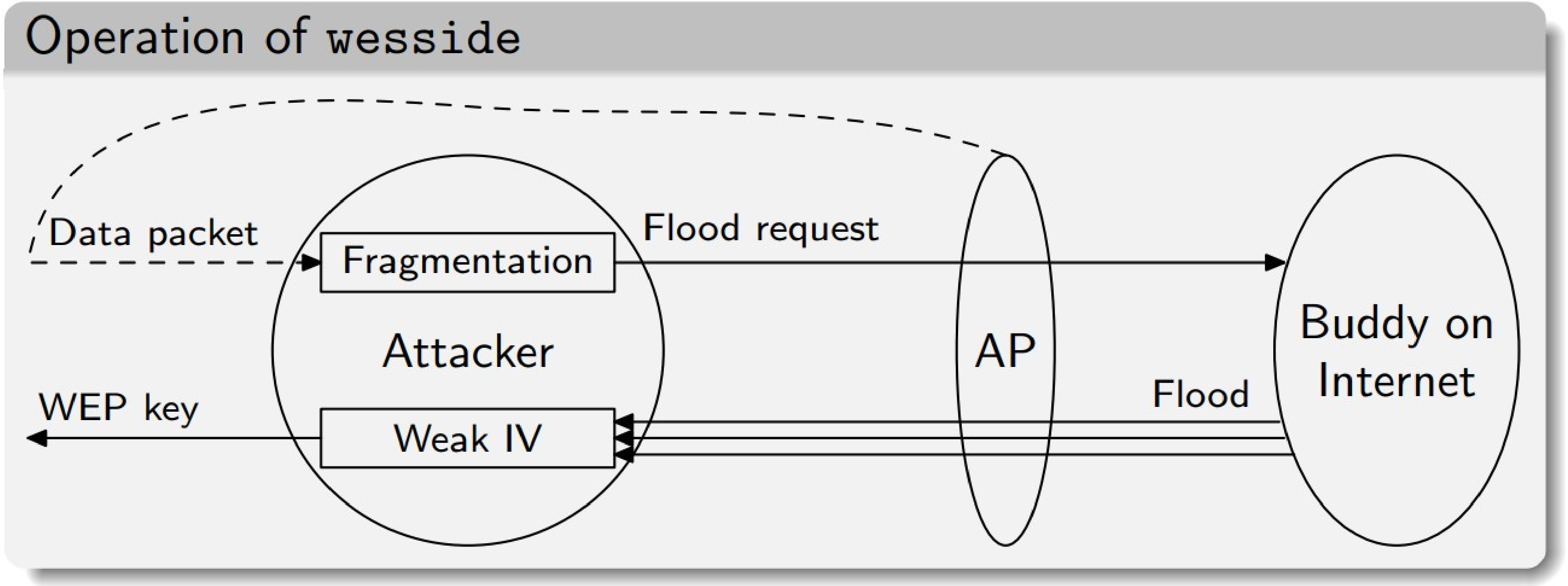
rc4-3.py

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

KRACK attacks...

https://www.youtube.com/watch?v=fZ1R9RliM1w https://papers.mathyvanhoef.com/ccs2017.pdf

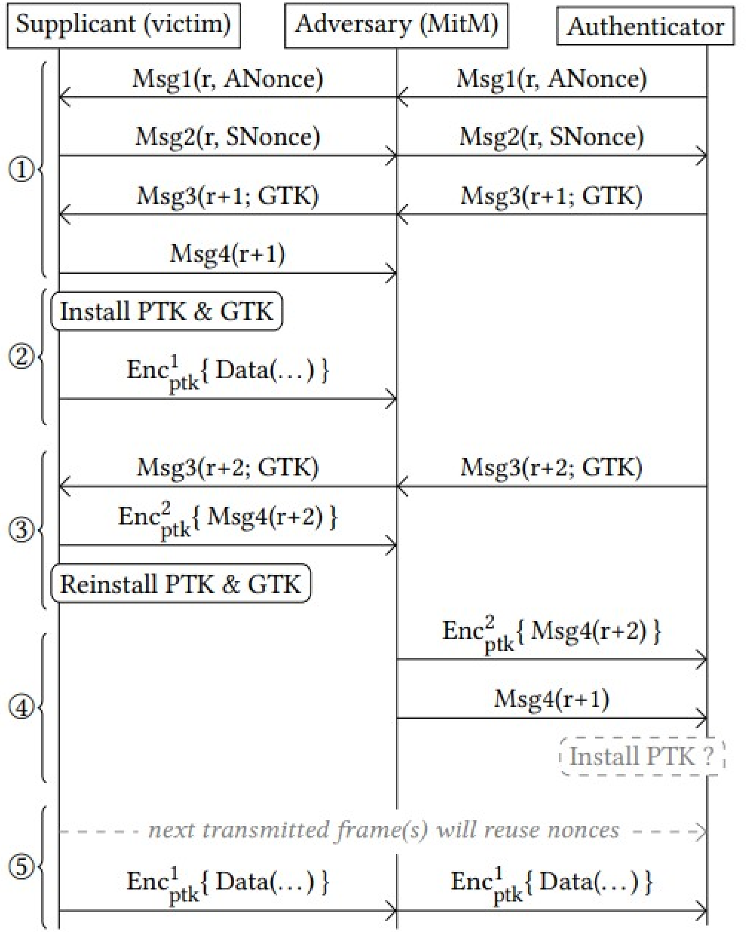

Supplicant (victim)
Adversary (MitM)
Authenticator
①
Msg1(r, ANonce)
Msg1(r, ANonce)
Msg2(r, SNonce)
Msg2(r, SNonce)
Msg3(r+1; GTK)
Msg3(r+1; GTK)
Msg4(r+1)
②
Install PTK & GTK
$\text{Enc}^{1}_{\text{ptk}}$\{ Data(...) \}
③
Msg3(r+2; GTK)
Msg3(r+2; GTK)
$\text{Enc}^{2}_{\text{ptk}}$\{ Msg4(r+2) \}
Reinstall PTK & GTK
④
$\text{Enc}^{2}_{\text{ptk}}$\{ Msg4(r+2) \}
Msg4(r+1)
Install PTK ?
⑤
next transmitted frame(s) will reuse nonces
$\text{Enc}^{1}_{\text{ptk}}$\{ Data(...) \}
$\text{Enc}^{1}_{\text{ptk}}$\{ Data(...) \}

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

KRACK attacks

Dragonblood attacks on WPA3

- Downgrade attacks (enterprise)
- Side channel (personal)
- Slides plagiarized from…

  https://papers.mathyvanhoef.com/wac2019-slides.pdf

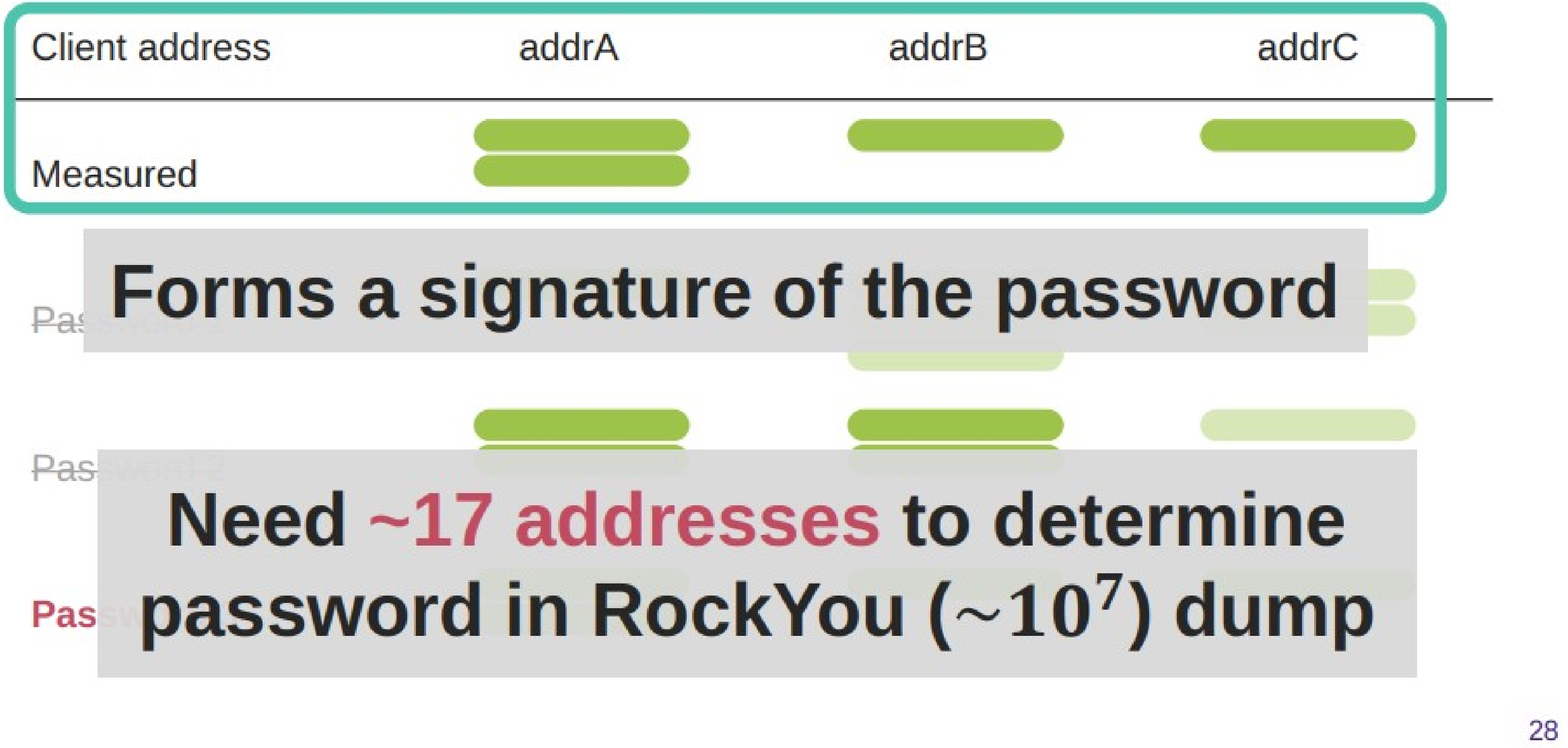
Convert password to MODP element

```
for (counter = 1; counter < 256; counter++)
    value = hash(pw, counter, addr1, addr2)
    if value >= p: continue
    P = $value^{(p-1)/q}$
    return P
```

Leaked information: #iterations needed

Client address	addrA	addrB	addrC
Measured	🟩 🟩	🟩	🟩
~~Password 1~~	🟩	░ ░ ░	░ ░
~~Password 2~~	🟩 🟩	🟩 🟩	░
Password 3	🟩 🟩	🟩	🟩

Leaked information: #iterations needed



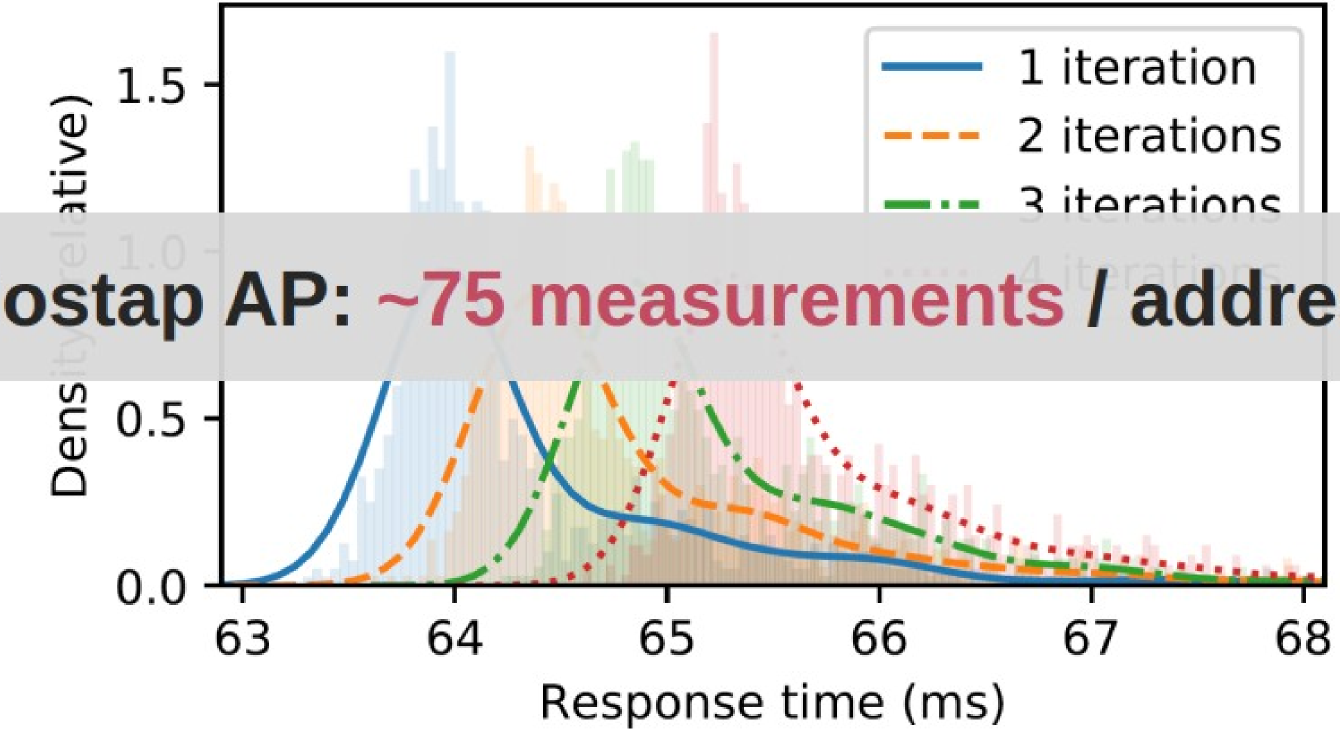
Client address	addrA	addrB	addrC
Measured	🟩 🟩	🟩	🟩
~~Password 1~~			
~~Password 2~~	🟩	🟩	🟩
Passw			

Forms a signature of the password

Need ~17 addresses to determine password in RockYou ($\sim 10^7$) dump

28

Raspberry Pi 1 B+: differences are measurable



Hostap AP: ~75 measurements / address

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:

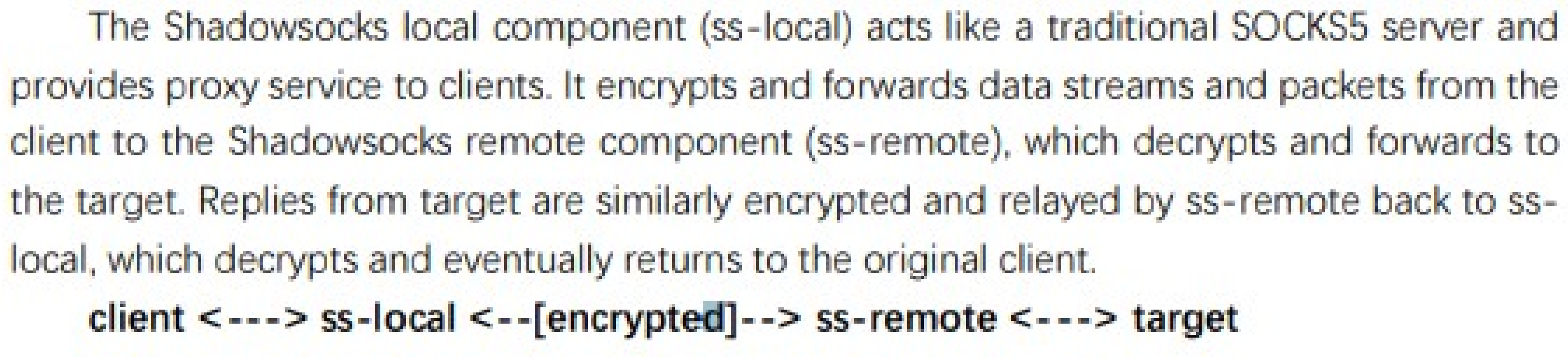
Steam ciphers (none-AEAD cipher):

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 defned:

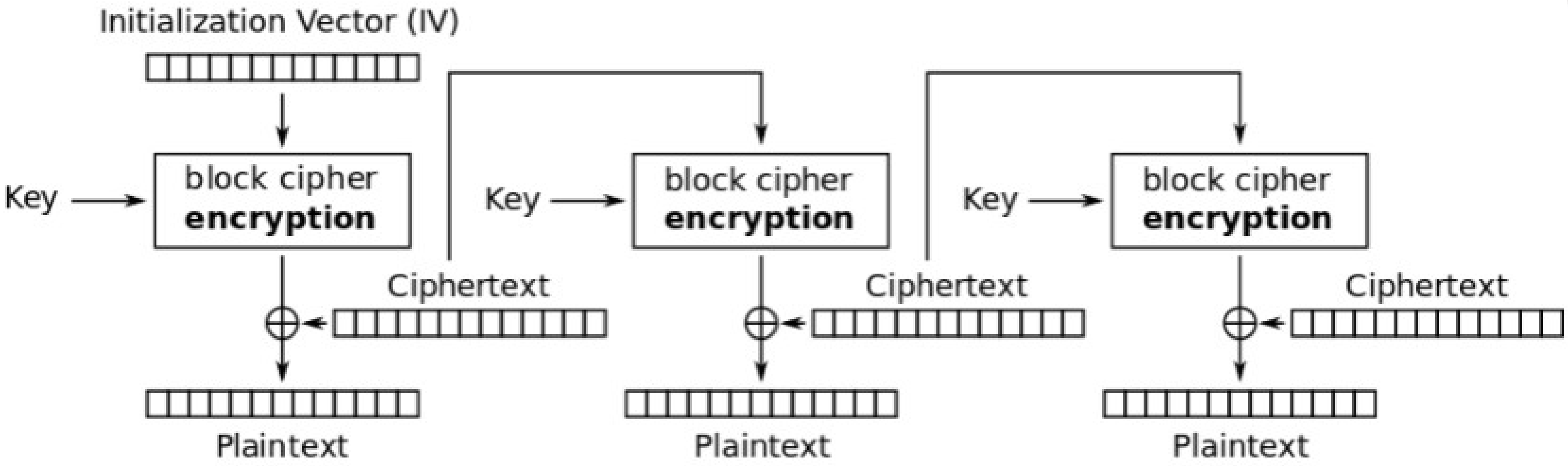
0x01: host is a 4-byte IPv4 address.

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

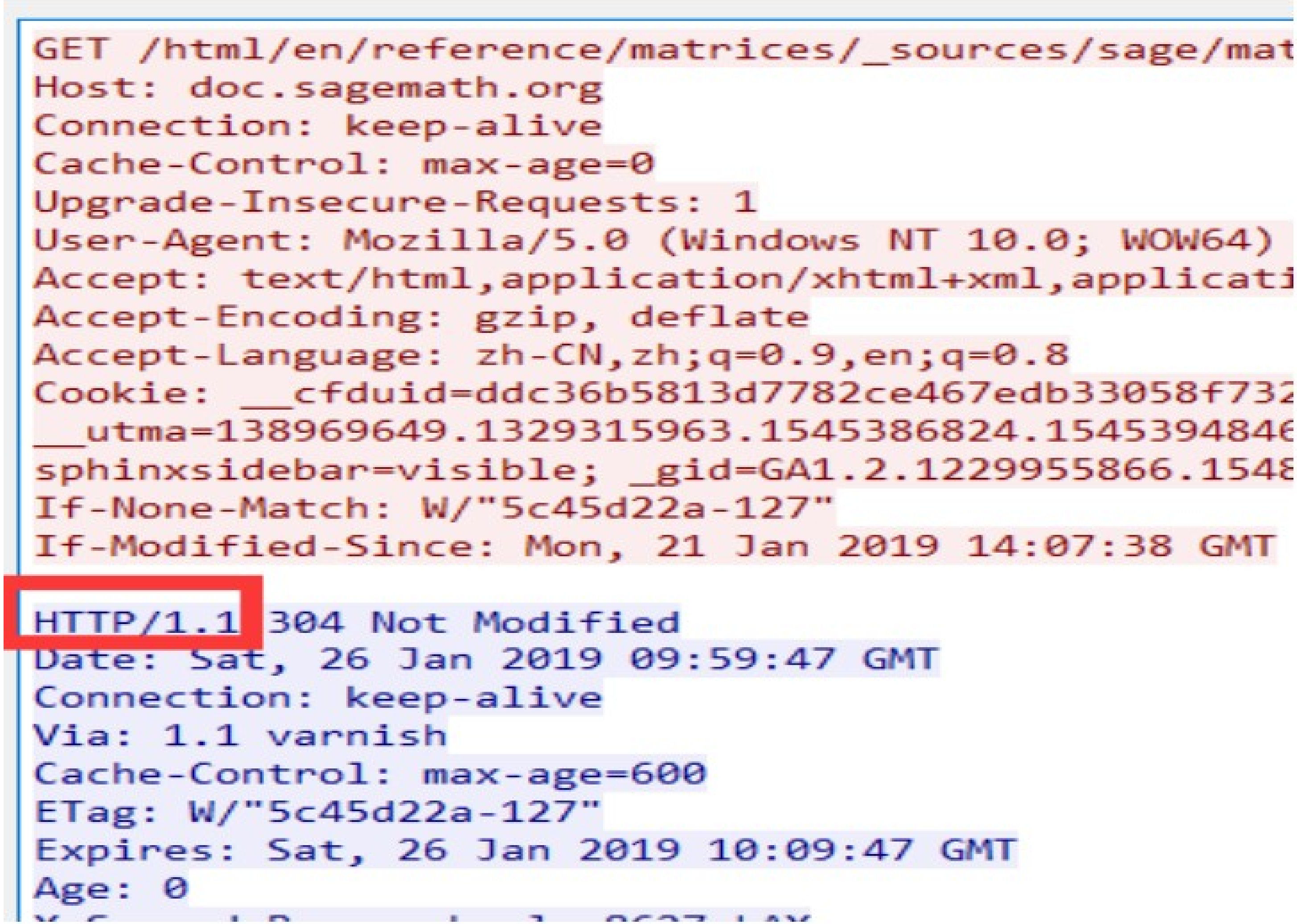
The port number is a 2-byte big-endian unsigned integer.

[IV][encrypted payload]

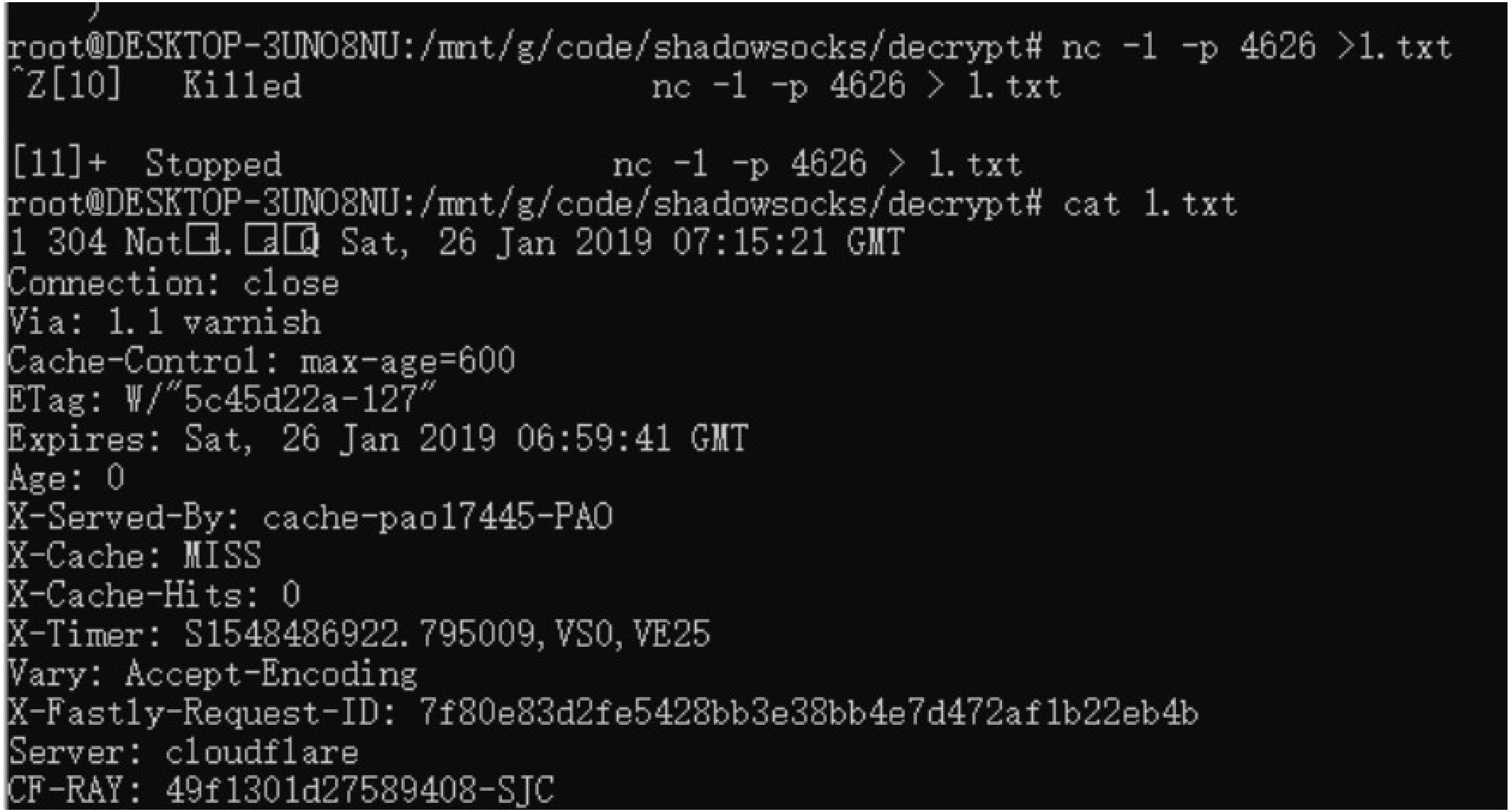


Cipher Feedback (CFB) mode decryption

IVs are chosen randomly, transmitted in plaintext.

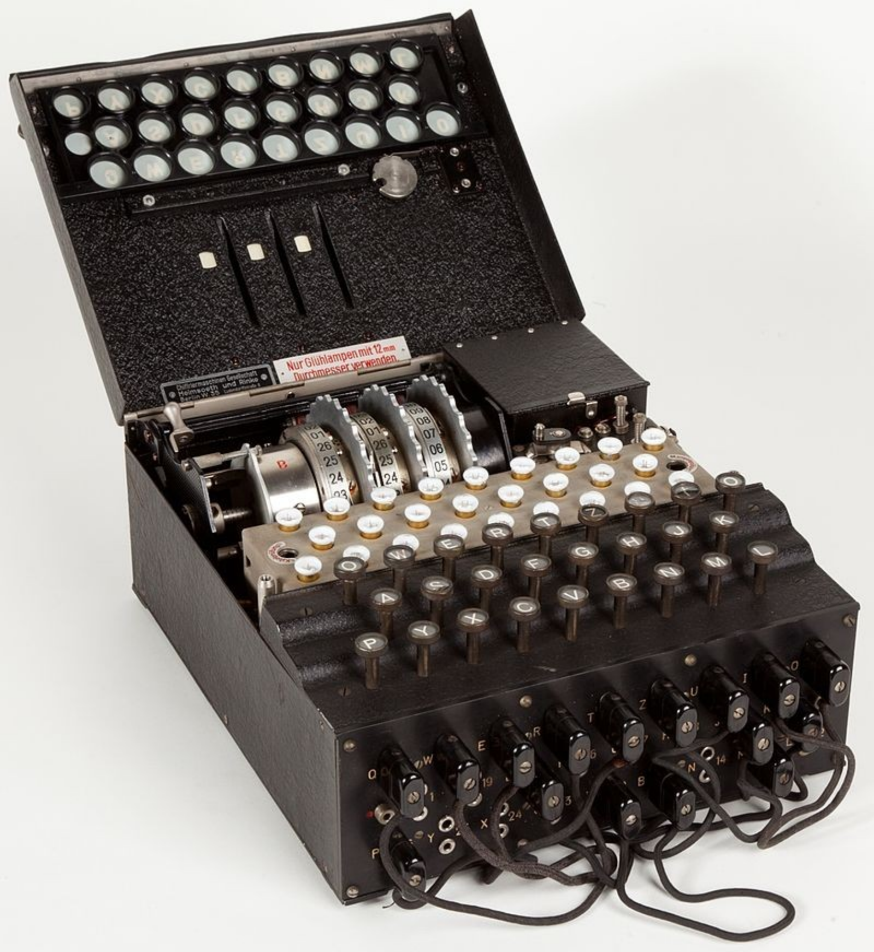


```
GET /html/en/reference/matrices/_sources/sage/mat
Host: doc.sagemath.org
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64)
Accept: text/html,application/xhtml+xml,applicati
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: __cfduid=ddc36b5813d7782ce467edb33058f732
__utma=138969649.1329315963.1545386824.154539484
sphinxsidebar=visible; _gid=GA1.2.1229955866.154
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
```

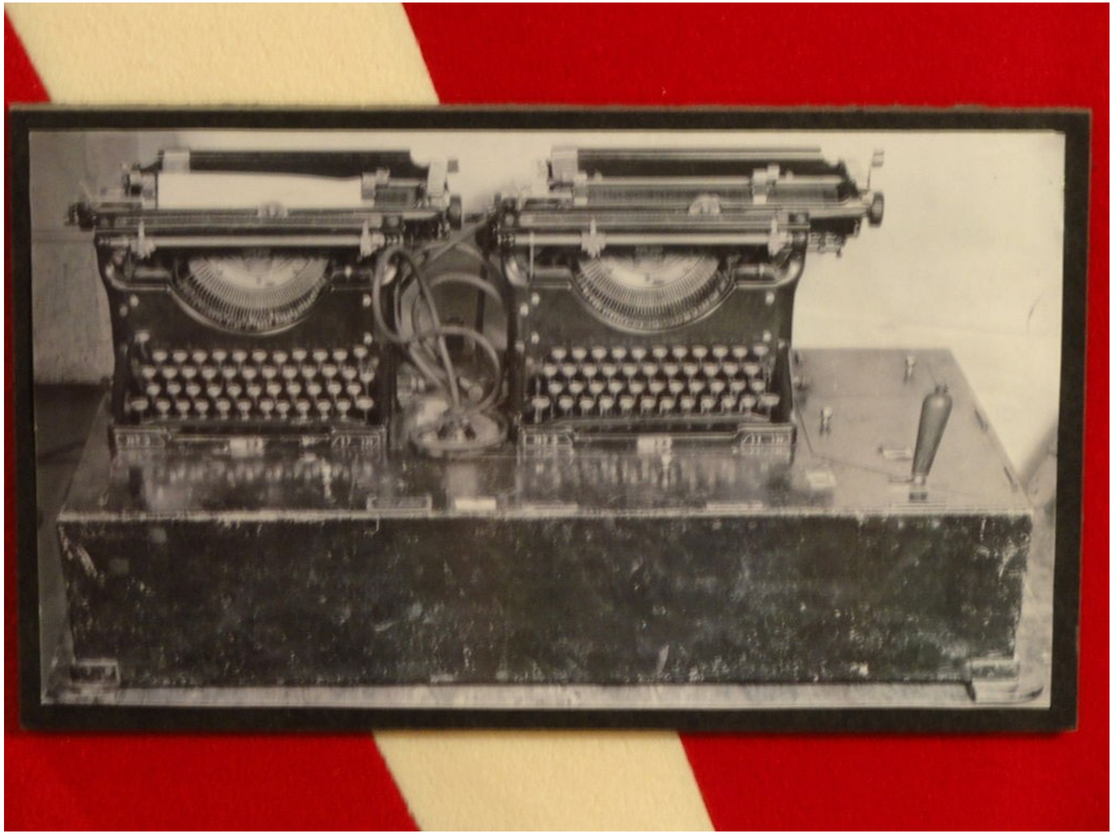


```
    )
root@DESKTOP-3UNO8NU:/mnt/g/code/shadowsocks/decrypt# nc -l -p 4626 >1.txt
^Z[10]   Killed                  nc -l -p 4626 > 1.txt

[11]+  Stopped                 nc -l -p 4626 > 1.txt
root@DESKTOP-3UNO8NU:/mnt/g/code/shadowsocks/decrypt# cat 1.txt
1 304 Not□1.□a□Q 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
```

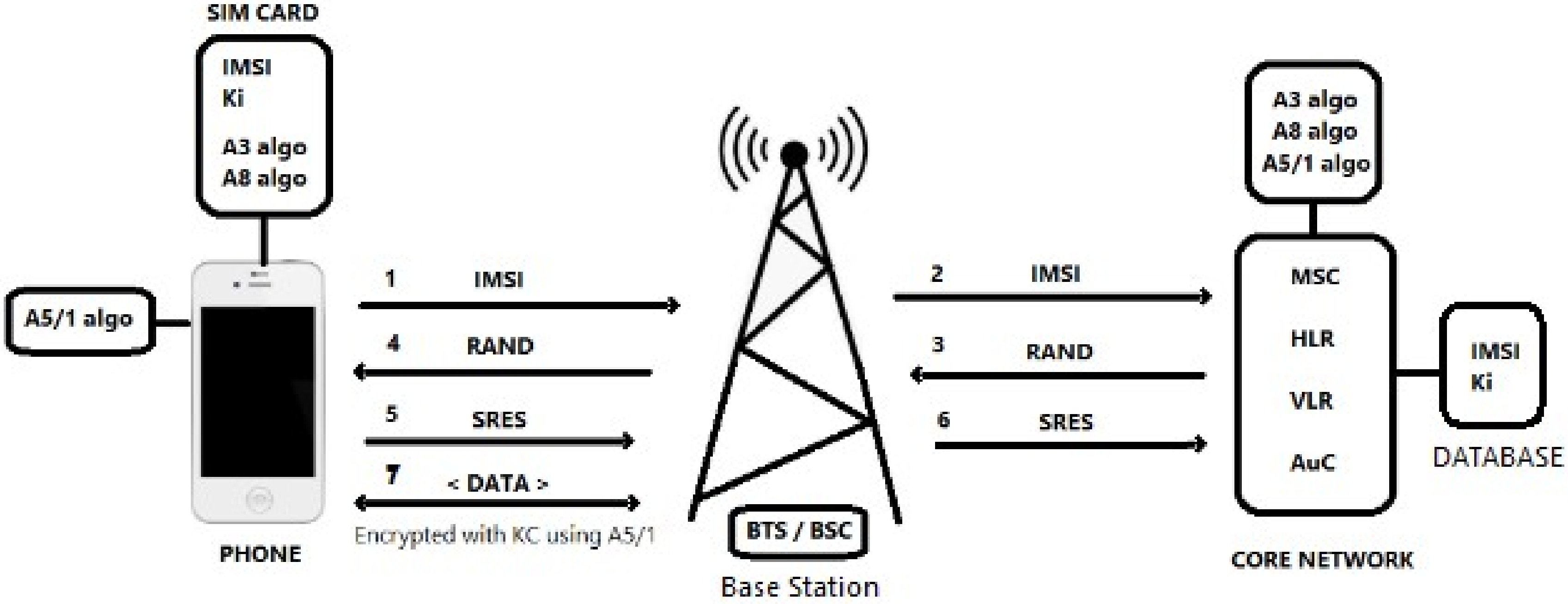
Many other stream cipher fails...



https://en.wikipedia.org/wiki/Enigma_machine#/media/File:Enigma_(crittografia)_-_Museo_scienza_e_tecnologia_Milano.jpg

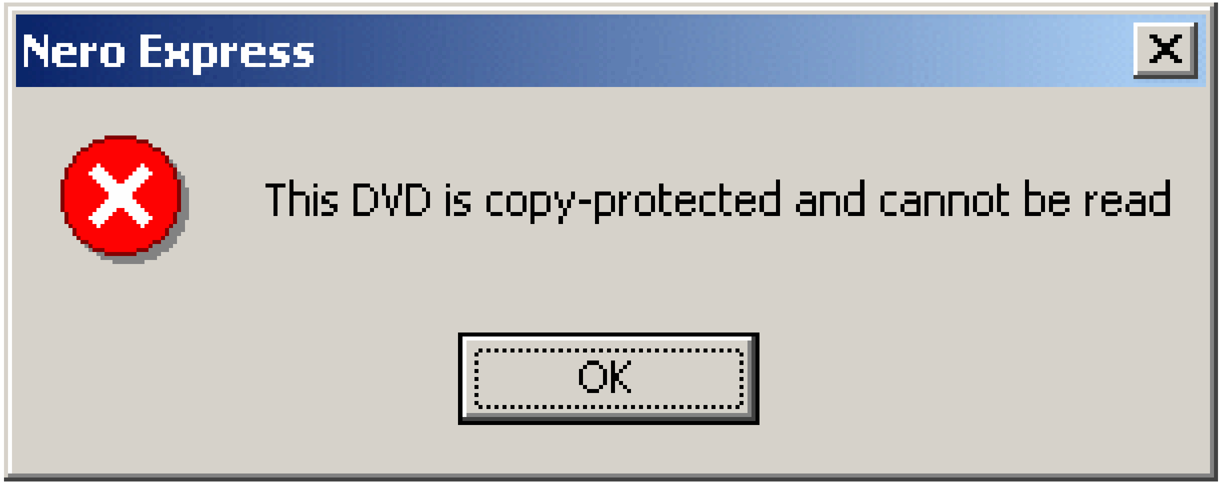


https://en.wikipedia.org/wiki/Type_B_Cipher_Machine#/media/File:Photograph_of_RED_cryptographic_device_-_National_Cryptologic_Museum_-_DSC07863.JPG



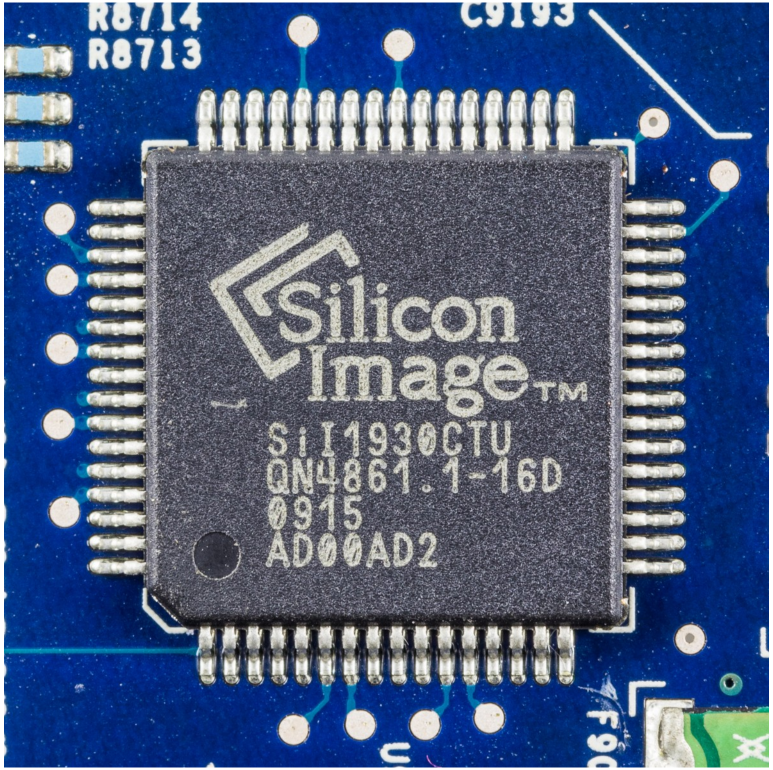
https://www.blackhillsinfosec.com/gsm-traffic-and-encryption-a5-1-stream-cipher/

Content Scramble System (CSS)



This DVD is copy-protected and cannot be read

High-bandwidth Digital Content Protection



https://commons.wikimedia.org/wiki/File:Apple_TV,_1st_generation_-_mainboard_-_Silicon_Image_SiI1930CTU-3215.jpg