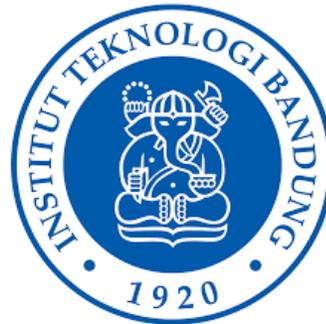


Bahan kuliah II4020 Kriptografi

# 07 - Serangan Terhadap Kriptografi



**Oleh: Rinaldi M**

**Program Studi Sistem dan Teknologi Informasi  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung  
2026**

# Pendahuluan

- Keseluruhan *point* dari kriptografi adalah menjaga kerahasiaan pesan atau kunci dari penyadap (*eavesdropper*) atau dari kriptanalis (*cryptanalyst*).
- Kriptanalis dapat pula merangkap sebagai seorang penyadap
- Kriptanalis berusaha memecahkan cipherteks dengan melakukan serangan terhadap sistem kriptografi.
- Tujuan serangan adalah untuk mengungkap plainteks dari cipherteks atau mendapatkan kunci dekripsi.

# Serangan (*attack*)

- **Serangan** diartikan sebagai setiap usaha (*attempt*) atau percobaan yang dilakukan oleh kriptanalis untuk menemukan kunci atau menemukan plainteks dari cipherteksnya.
- Asumsi: kriptanalis mengetahui algoritma kriptografi yang digunakan

**Prinsip Kerckhoff:** Semua algoritma kriptografi harus publik; hanya kunci yang rahasia.

Jadi, keamanan sistem kriptografi terletak pada **kunci!**

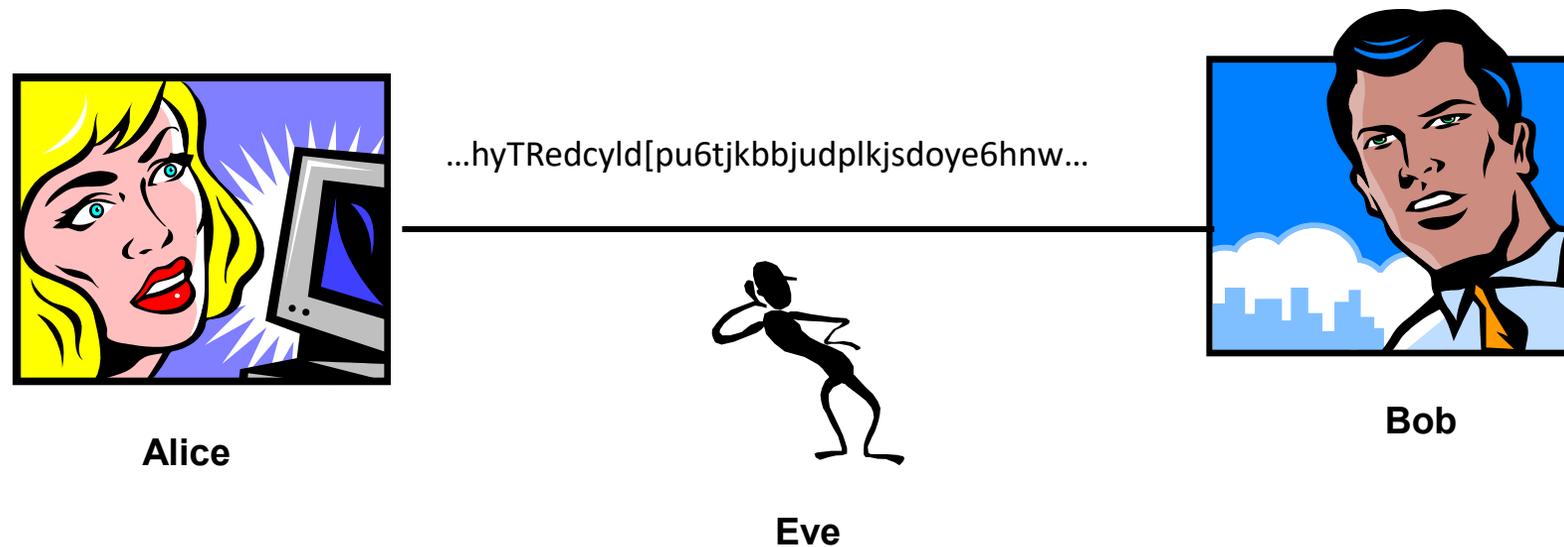
# Jenis-jenis Serangan

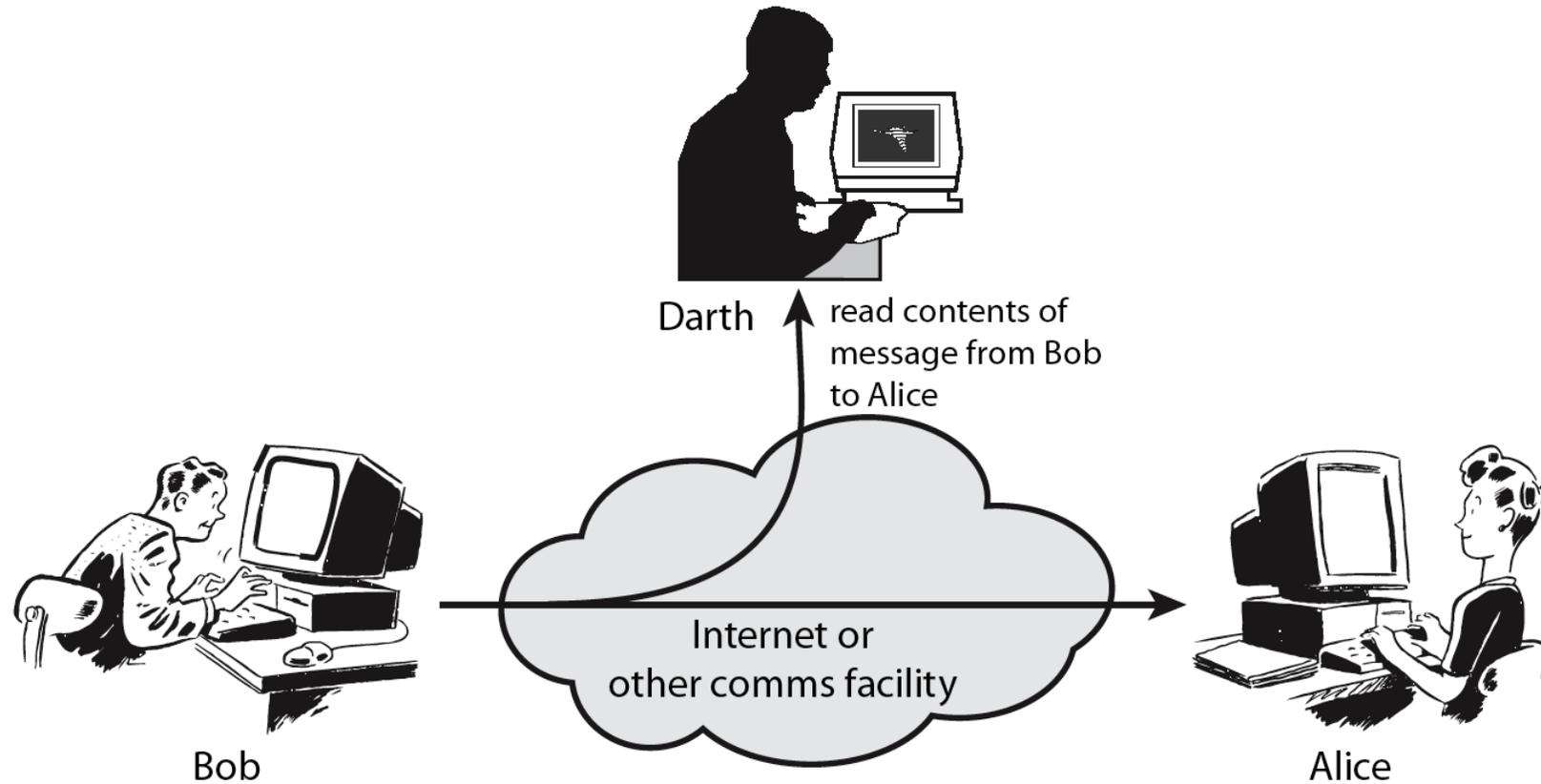
Berdasarkan keterlibatan penyerang dalam komunikasi pesan:

- 1. Serangan pasif**
- 2. Serangan aktif**

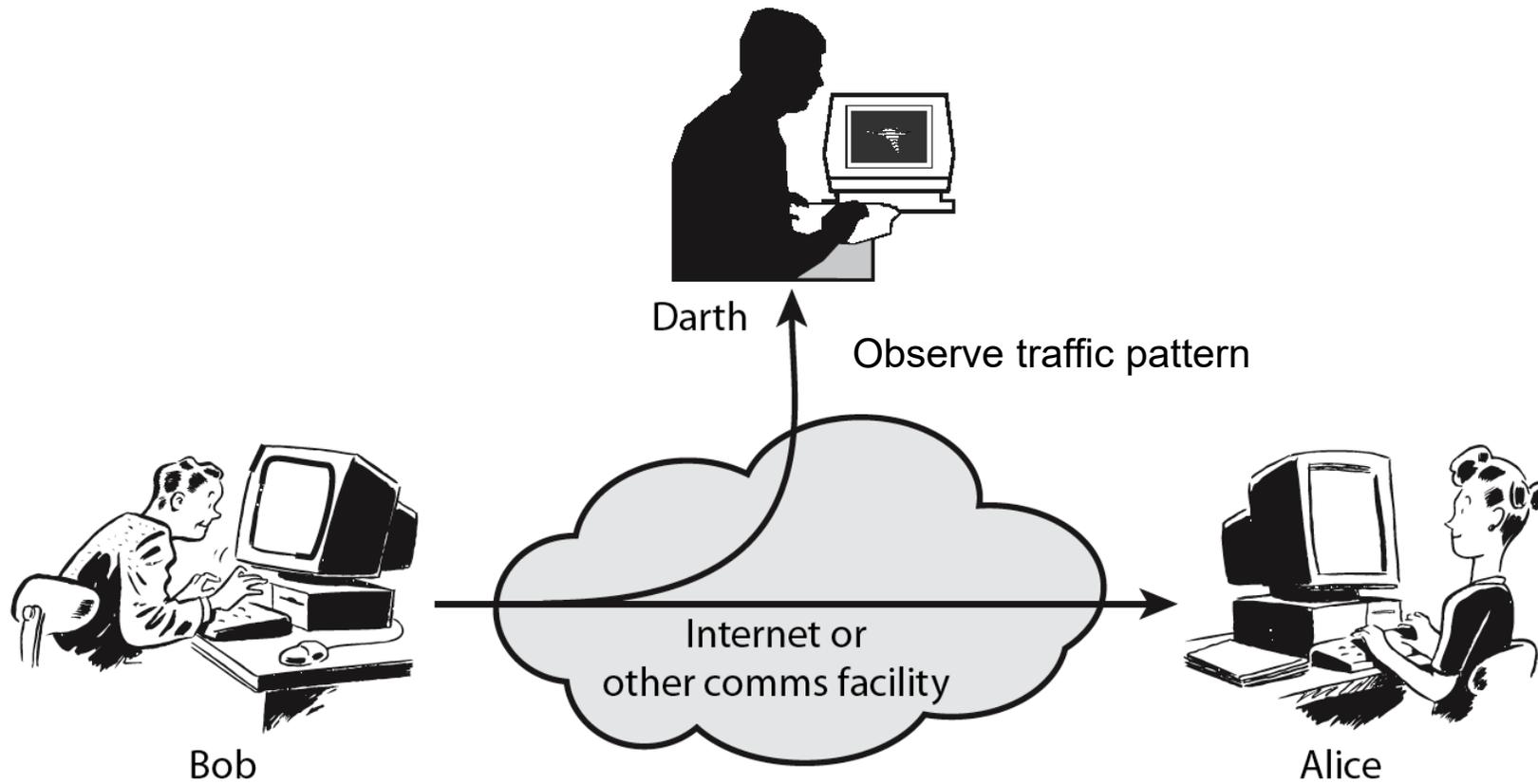
# 1. Serangan pasif (*passive attack*)

- penyerang tidak terlibat dalam komunikasi antara pengirim dan penerima
- penyerang hanya melakukan penyadapan untuk memperoleh data atau informasi sebanyak-banyaknya
- enkripsi pesan mencegah penyadap memahami isi pesan





## Passive Attack : Interception



## Passive Attack : Traffic Analysis

# Screenshot Wireshark (memantau network traffic)

The screenshot shows the Wireshark interface capturing traffic from a Marvell Yukon Ethernet Controller. The main window displays a list of captured packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The selected packet (No. 27) is highlighted in blue. Below the list, the packet details pane shows the structure of the selected packet: Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0. The details include IEEE 802.3 Ethernet, Logical-Link Control, Internetnetwork Packet exchange, and NetBIOS over IPX. The packet bytes pane shows the raw data in hexadecimal and ASCII format.

No.	Time	Source	Destination	Protocol	Length	Info
27	2.79561600	167.205.33.90	255.255.255.255	DB-LSP-	154	Dropbox LAN sync Discovery Protocol
28	2.96394400	167.205.33.14	167.205.33.255	NBNS	92	Name query NB CORPORATE<00>
29	3.06038000	00000000.00804837fc	00000000.ffffffff	NBIPX	98	Find name WORKGROUP<00>
30	3.06039600	4416db43.0000000000	00000000.00804837fc	NBIPX	98	Name recognized WORKGROUP<00>
31	3.06809700	167.205.33.14	167.205.33.255	NBNS	92	Name query NB CORPORATE<00>
32	3.09102200	167.205.33.88	167.205.33.255	NBNS	92	Name query NB PRINTERBASDAD<00>
33	3.24244300	AsustekC_10:09:66	Broadcast	ARP	60	who has 167.205.33.12? Tell 167.205.33.2
34	3.30945700	Cisco-Li_11:0d:0f	Spanning-tree-(for- STP	STP	60	RST. Root = 32768/0/00:22:6b:10:d8:3d Co
35	3.35738600	167.205.33.121	167.205.33.255	NBNS	92	Name query NB SUDARMAN<20>
36	3.47038100	167.205.33.61	50.62.3.118	TCP	62	mctet-gateway > https [SYN] Seq=0 win=655
37	3.60684800	IntelCor_c2:e0:11	Broadcast	ARP	60	who has 167.205.33.116? Tell 167.205.33.
38	3.71257800	167.205.33.14	167.205.33.255	NBNS	92	Name query NB CORPORATE<00>
39	3.80628200	167.205.33.14	167.205.33.255	NBNS	92	Name query NB CORPORATE<00>
40	3.83975500	00000000.00804837fc	00000000.ffffffff	BROWSEF	176	Request Announcement DAPUR
41	3.83996700	167.205.33.100	167.205.33.255	BROWSEF	243	Host Announcement BUGI-WIBOWO, workstatio

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

- IEEE 802.3 Ethernet
- Logical-Link Control
- Internetnetwork Packet exchange
- NetBIOS over IPX

```
0000 ff ff ff ff ff ff 00 80 48 37 fc 30 00 54 e0 e0 ..... H7.0.T..
0010 03 ff ff 00 50 00 14 00 00 00 00 ff ff ff ff ff ....P...
0020 ff 04 55 00 00 00 00 00 80 48 37 fc 30 04 55 00 ..U.....H7.0.U.
0030 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 c0 .....
0050 02 01 02 5f 5f 4d 52 42 52 4f 57 52 45 5f 5f 02 ..... MSB POWER
```

http.request.method=="POST"

No.	Time	Source	Destination	Protocol	Length	Info
1034	8.148165	172.99.96.253	160.153.129.234	HTTP	617	POST /sign

[Full request URI: <http://www.sababank.com/signin.php>]  
 [HTTP request 1/1]  
 [Response in frame: 1129]  
 File Data: 53 bytes

- HTML Form URL Encoded: application/x-www-form-urlencoded
  - Form item: "username" = "Ibrahim\_Diyeb"
  - Form item: "password" = "yemen\_123"
  - Form item: "actn" = "signin"

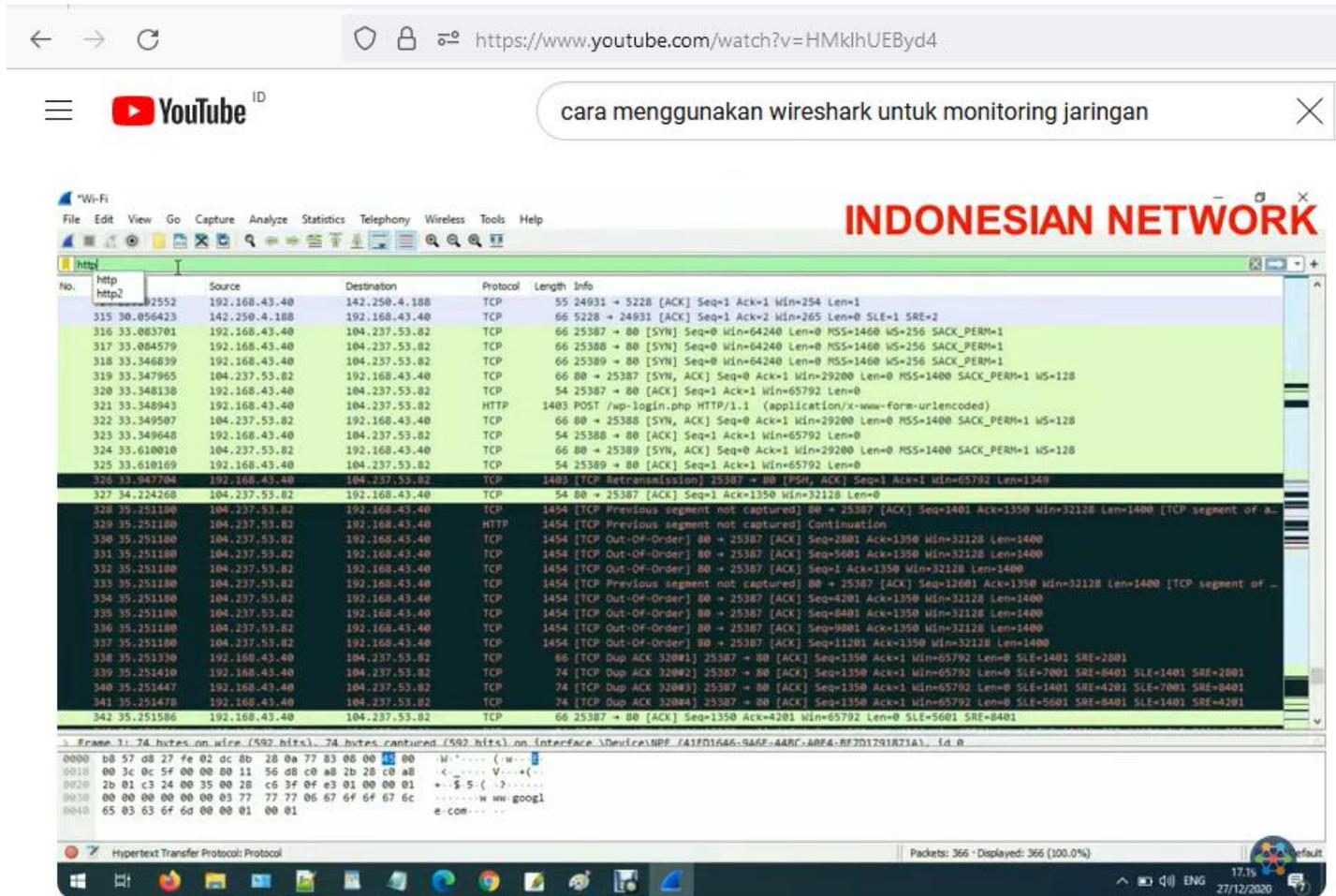
01a0	63 6f 64 65 64 0d 0a 43	6f 6e 74 65 6e 74 2d 4c	coded..Content-L
01b0	65 6e 67 74 68 3a 20 35	33 0d 0a 43 6f 6f 6b 69	ength: 5 3..Cooki
01c0	65 3a 20 50 48 50 53 45	53 53 49 44 3d 34 31 32	e: PHPSESSID=412
01d0	33 35 34 31 32 30 63 35	36 37 34 35 61 63 66 34	354120c5 6745acf4
01e0	31 62 38 65 32 39 36 34	63 32 62 65 35 3b 20 6c	1b8e2964 c2be5; l
01f0	61 6e 67 3d 61 72 61 62	69 63 0d 0a 43 6f 6e 6e	ang=arabic..Conn
0200	65 63 74 69 6f 6e 3a 20	6b 65 65 70 2d 61 6c 69	ection: keep-ali
0210	76 65 0d 0a 55 70 67 72	61 64 65 2d 49 6e 73 65	ve..Upgrade-Inse
0220	63 75 72 65 2d 52 65 71	75 65 73 74 73 3a 20 31	cure-Requests: 1
0230	0d 0a 0d 0a 75 73 65 72	6e 61 6d 65 3d 49 62 72	...user name=Ibr
0240	61 68 69 6d 5f 44 69 79	65 62 26 70 61 73 73 77	ahim_Diyeb&passw

Filtering dengan Wireshark dapat menampilkan plainteks berupa *username* dan *password*

Sumber gambar: [https://www.researchgate.net/figure/Wireshark-Filtering-Showing-Clear-Text-of-user-Name-and-Password\\_fig3\\_326419957](https://www.researchgate.net/figure/Wireshark-Filtering-Showing-Clear-Text-of-user-Name-and-Password_fig3_326419957)

# Cek video TUTORIAL CARA MENGGUNAKAN WIRESHARK | MENEMUKAN U53RN4M3 P455WORD PADA LALU LINTAS PACKET JARINGAN:

<https://www.youtube.com/watch?v=HMklhUEByd4>



## TUTORIAL CARA MENGGUNAKAN WIRESHARK | MENEMUKAN U53RN4M3 P455WORD PADA LALU LINTAS PACKET JARINGAN

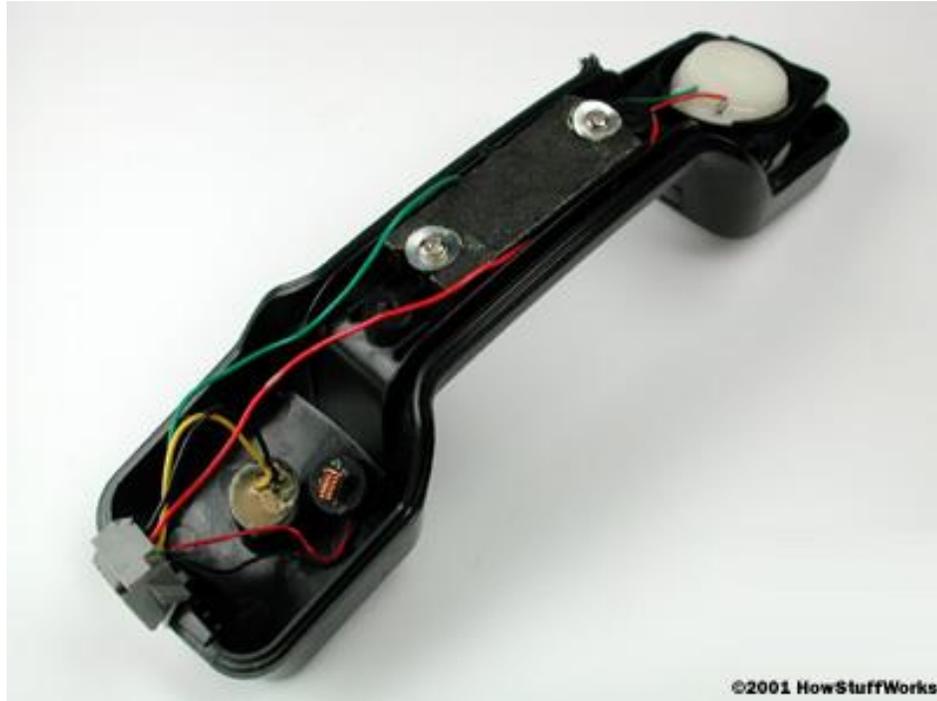
## **Metode penyadapan:**

1. *Wiretapping*
2. *Electromagnetic eavesdropping*
3. *Acoustic Eavesdropping*



# How Wiretapping Works

(sumber: <http://www.howstuffworks.com/wiretapping.htm>)

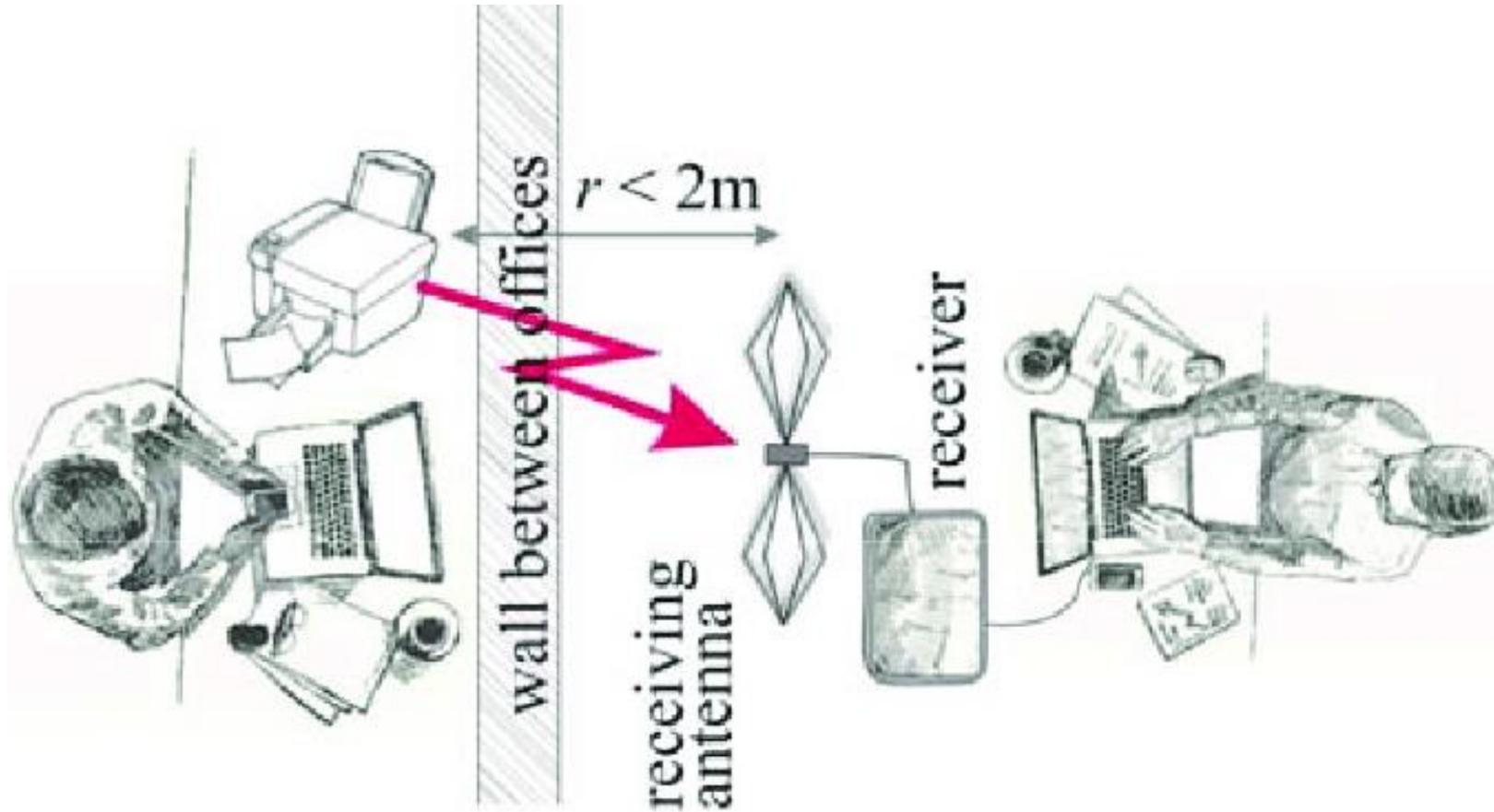


**When you open up a phone, you can see that the technology inside is very simple. The simplicity of design makes the phone system vulnerable to surreptitious eavesdropping.**



**Inside a standard phone cord, you'll find a red wire and a green wire. These wires form a circuit like the one you might find in a flashlight. Just as in a flashlight, there is a negatively-charged end and a positively-charged end to the circuit. In a telephone cord, the green wire connects to the positive end and the red cord connects to the negative end.**

# Electromagnetic eavesdropping



Sumber: [https://www.researchgate.net/figure/An-anechoic-chamber-Figure-4-Example-for-an-electromagnetic-eavesdropping-process\\_fig3\\_324680618](https://www.researchgate.net/figure/An-anechoic-chamber-Figure-4-Example-for-an-electromagnetic-eavesdropping-process_fig3_324680618)

Cek video **INTERCEPT ANY RADIO SIGNAL!!!!**: <https://www.youtube.com/watch?v=UMu5AhSg-TI>



The screenshot shows a YouTube video player interface. At the top, there is a browser address bar with the URL <https://www.youtube.com/watch?v=UMu5AhSg-TI>. Below the address bar is the YouTube logo and a search bar containing the text "Telusuri". The main video frame shows a man with a beard and a black shirt holding a small black device with a screen. The screen displays a yellow waveform on a black background, resembling a spectrum analyzer or a signal capture tool. The background of the video shows a computer monitor with various data and graphs.

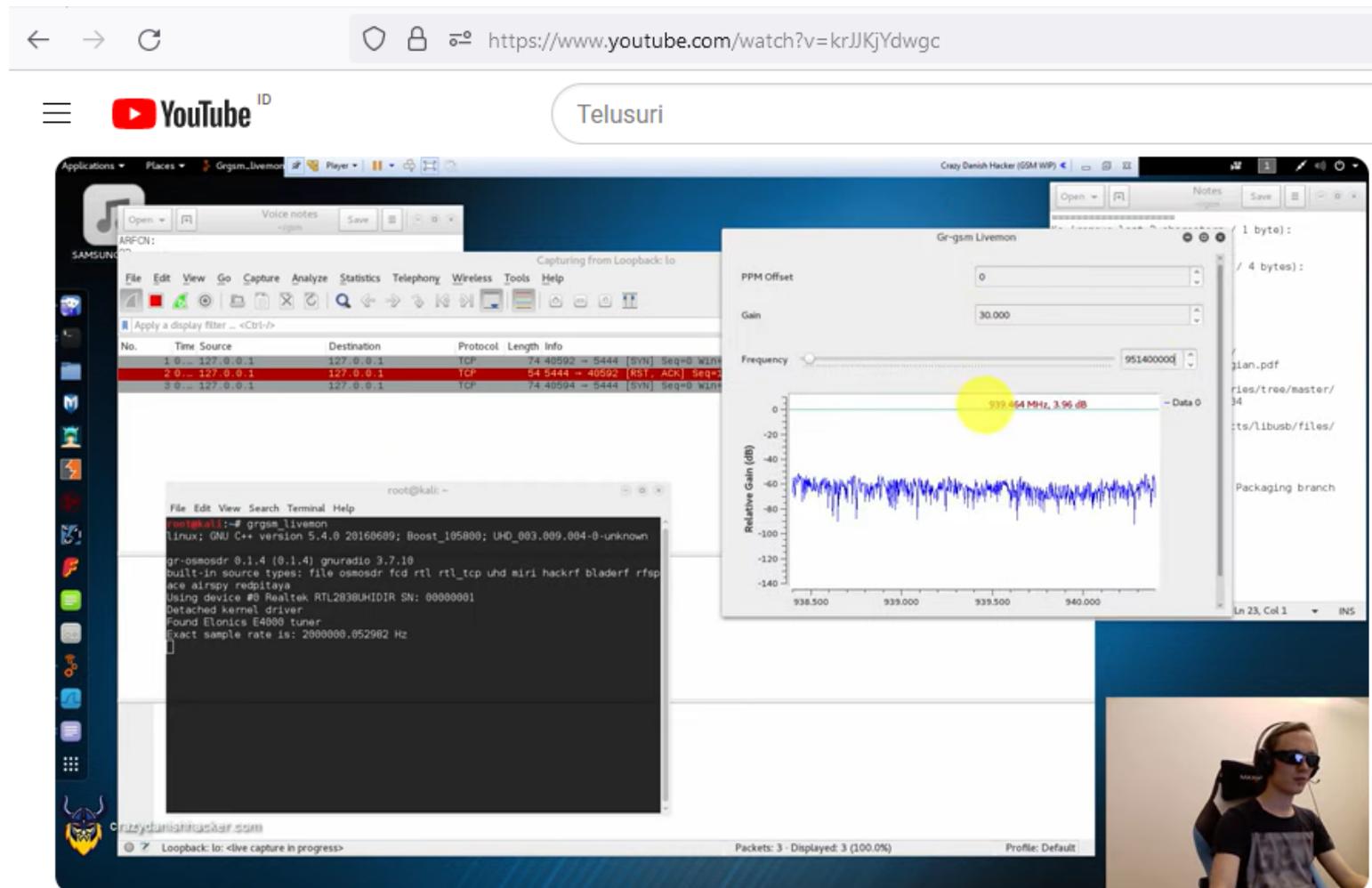
**INTERCEPT ANY RADIO SIGNAL!!!!**

 **andy kirby**  
105 rb subscriber

[Subscribe](#)

 7,4 rb  [Bagikan](#) 

# Cek video GSM Sniffing: Voice Decryption 101 - Software Defined Radio Series #11: <https://www.youtube.com/watch?v=krJJKjYdwgc>



## GSM Sniffing: Voice Decryption 101 - Software Defined Radio Series #11



**Crazy Danish Hacker**  
26,9 rb subscriber

Subscribe

2 rb



Bagikan

Download



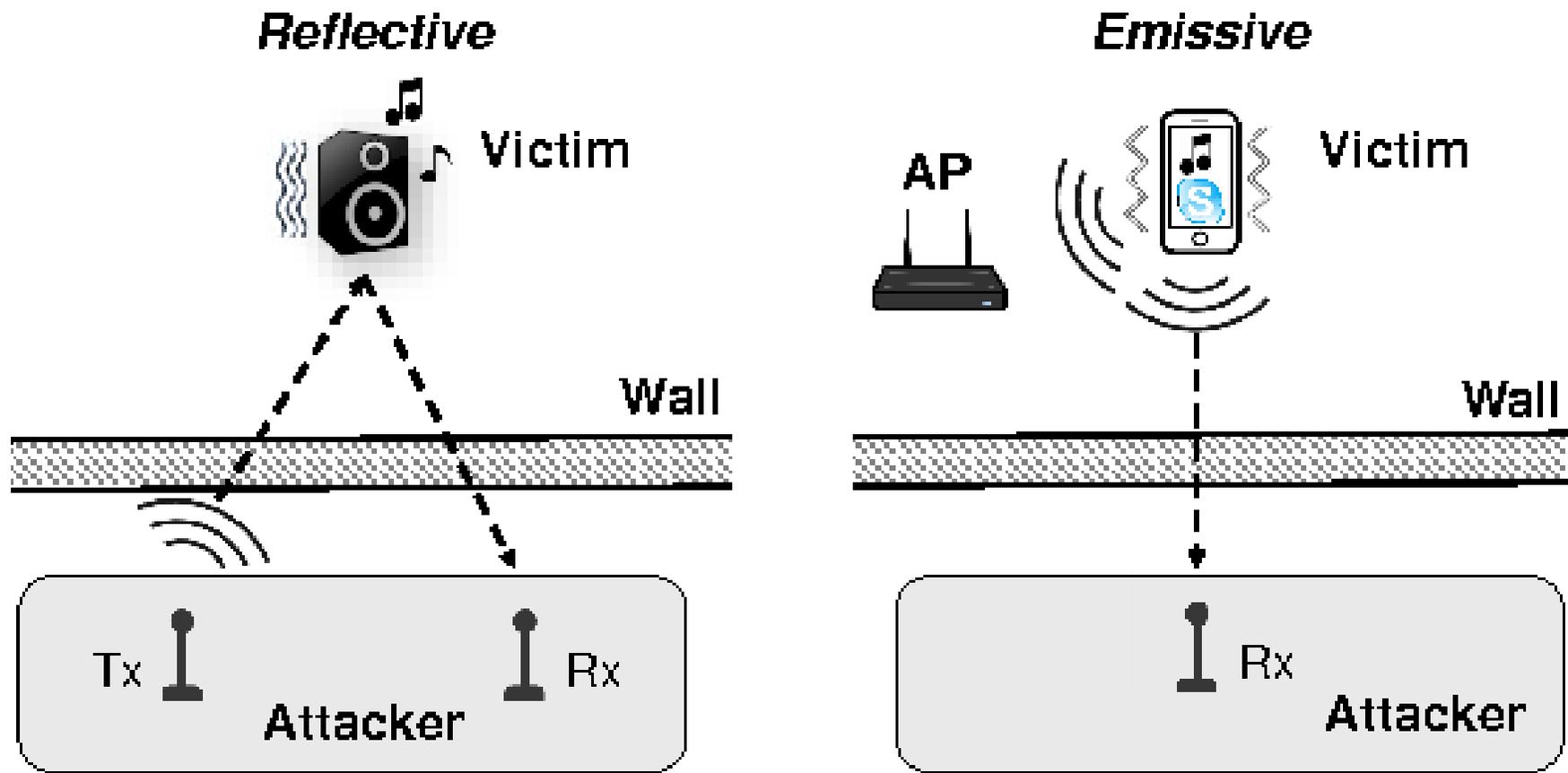
# Acoustic Eavesdropping



15506-41DG  
'Office: 9am' Disc  
© JupiterImages

Creatas

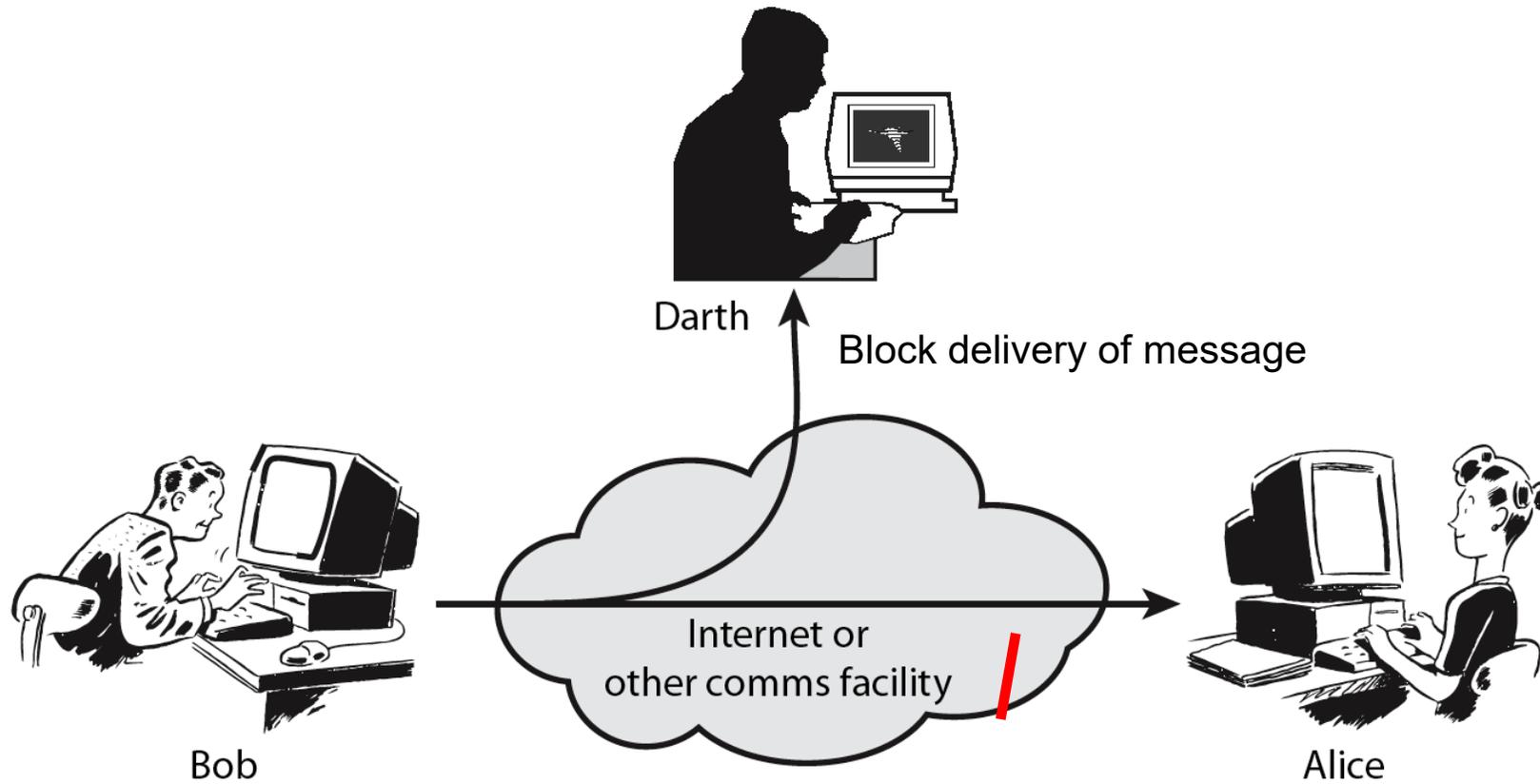
[www.comstock.com](http://www.comstock.com)



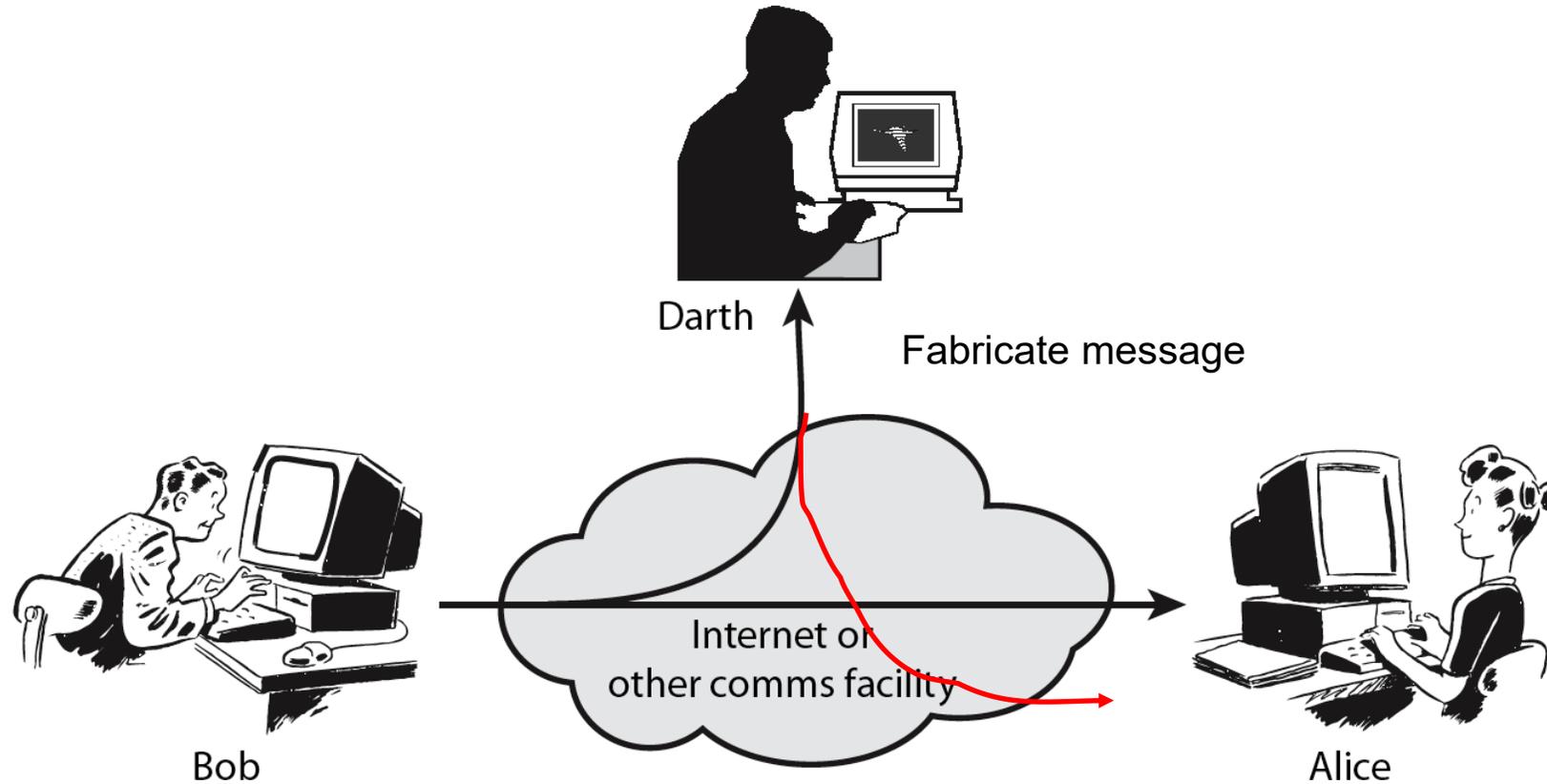
Sumber: <https://www.semanticscholar.org/paper/Acoustic-Eavesdropping-through-Wireless-Vibrometry-Wei-Wang/8afd80726c54ed7b95d30d1230bef633d128c930>

## 2. Serangan aktif (*active attack*)

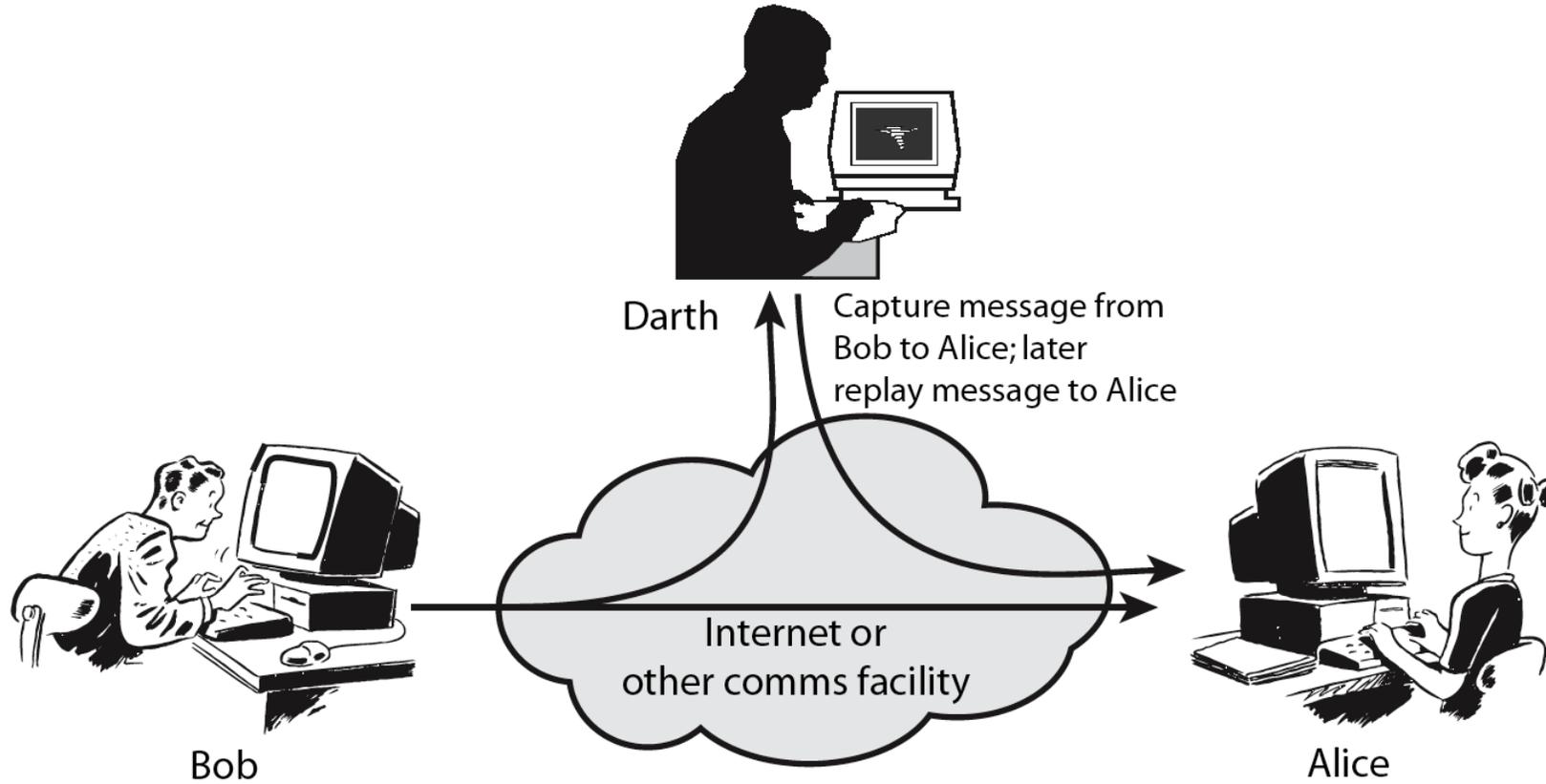
- penyerang mengintervensi komunikasi dan ikut mempengaruhi sistem untuk keuntungan dirinya
- penyerang dapat mengubah aliran pesan seperti:
  - menghapus sebagian cipherteks,
  - mengubah cipherteks,
  - menyisipkan potongan cipherteks palsu,
  - *me-replay* pesan lama,
  - mengubah informasi yang tersimpan, dsb
- Contoh: *man-in-the-middle attack*



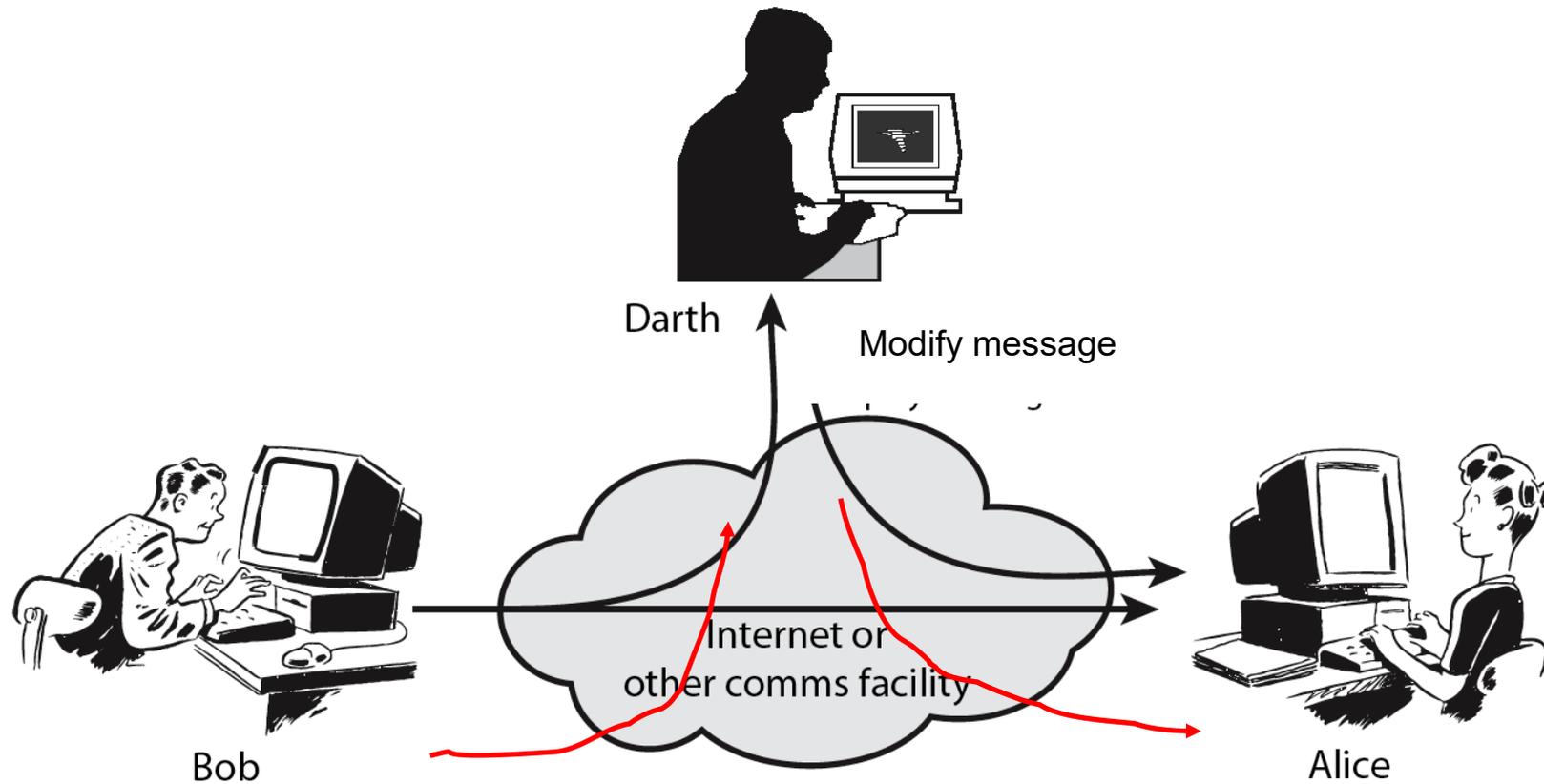
## Active Attack: Interruption



## Active Attack: Fabrication

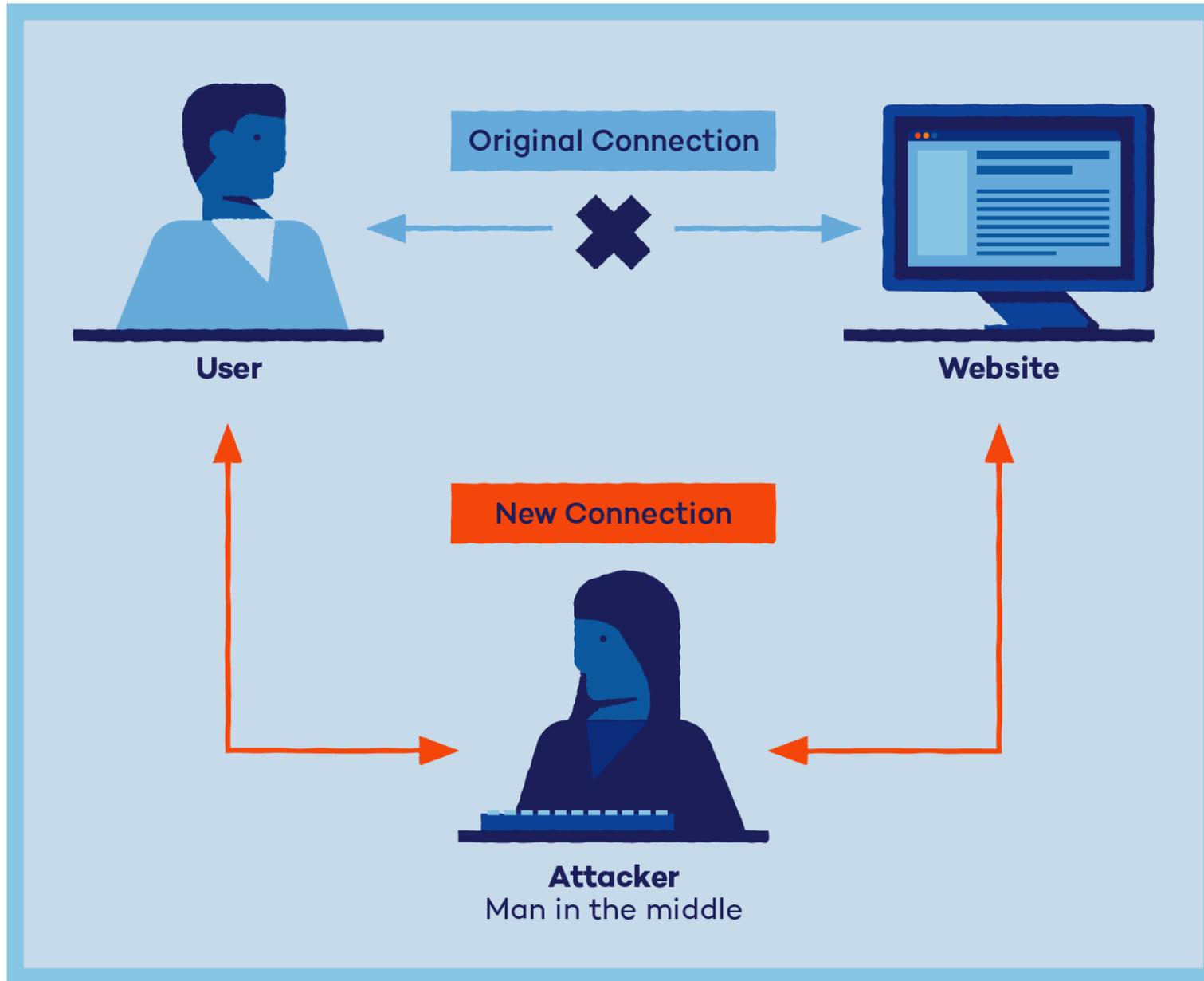


## Active Attack: Replay

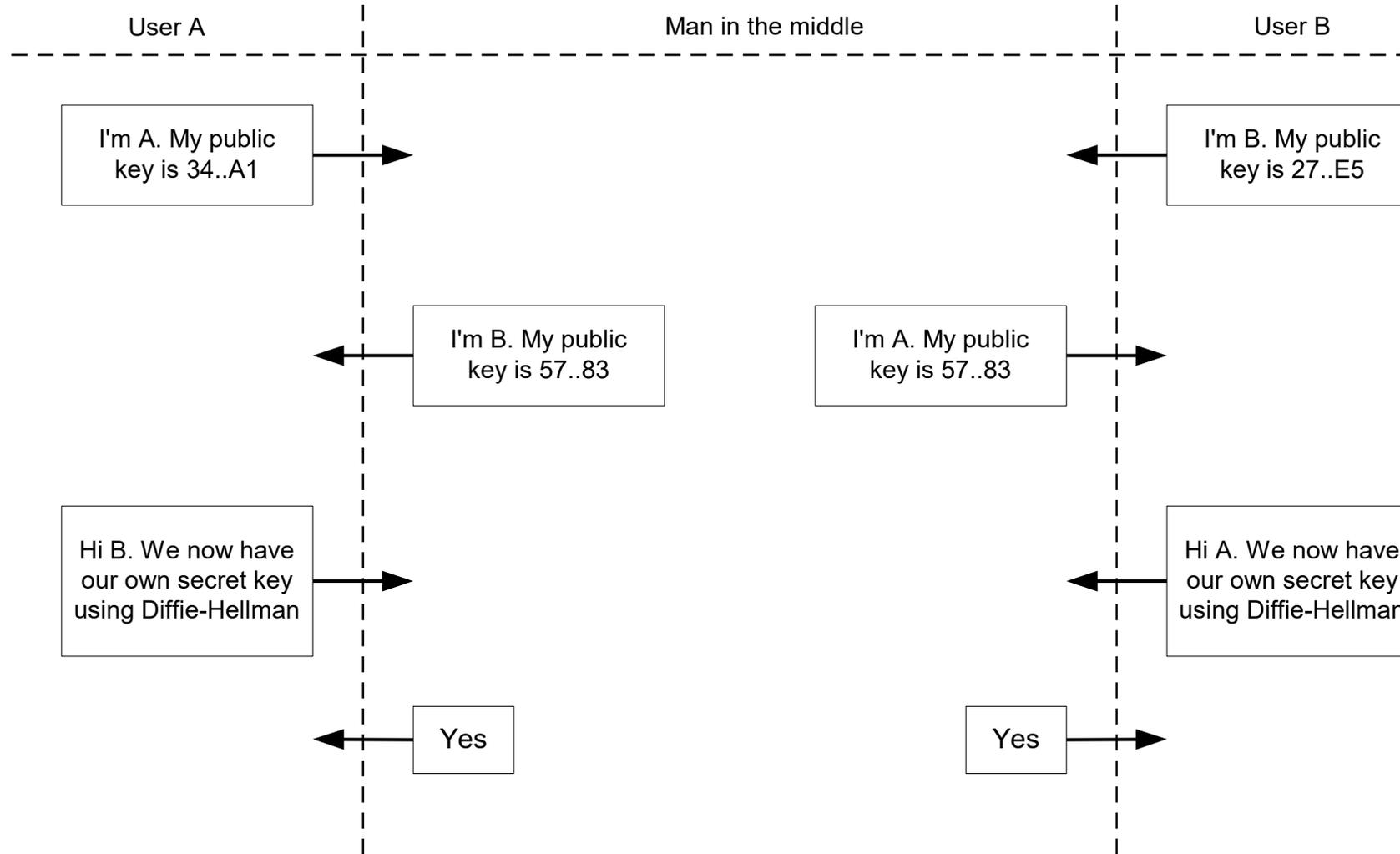


## Active Attack: Modification

# *Man-in-the-middle-attack*

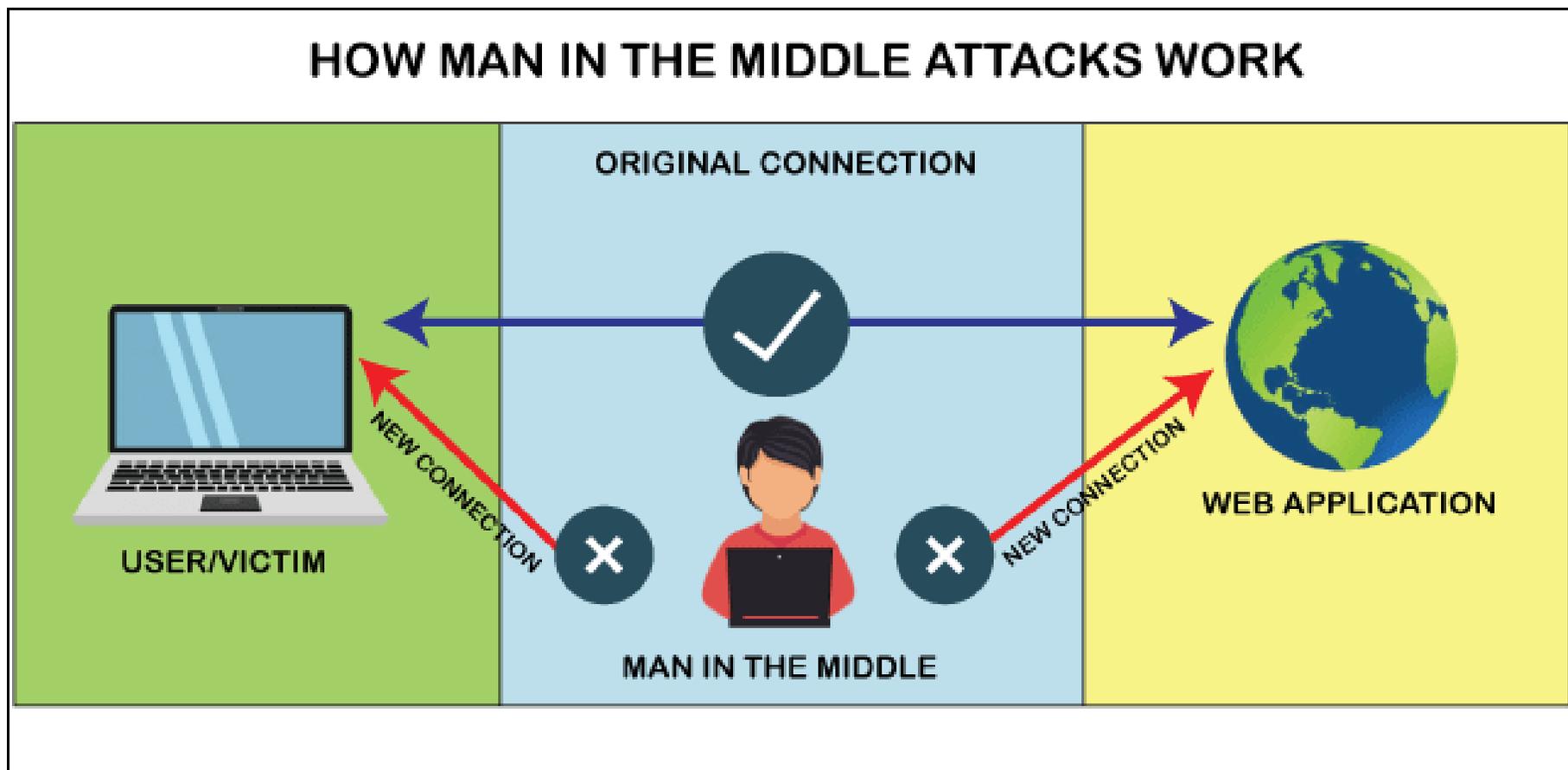


# Man-in-the-middle-attack



# *Man-in-the-middle-attack*

Serangan aktif yang berbahaya



# Jenis-jenis Serangan

Berdasarkan teknik yang digunakan untuk menemukan kunci:

1. *Exhaustive attack/brute force attack*
2. *Analytical attack*

## ***1. Exhaustive attack /brute force attack***

- Mengungkap plainteks dengan mencoba semua kemungkinan kunci untuk dekripsi cipherteks
- . Contoh: *dictionary attack*
- Pasti berhasil menemukan kunci jika diberikan waktu yang cukup dan sumberdaya *hardware* dan *software* yang memenuhi.

**Tabel 1** Waktu yang diperlukan untuk *exhaustive key search*  
 (Sumber: William Stallings, *Data and Computer Communication Fourth Edition*)

Ukuran kunci	Jumlah kemungkinan kunci	Lama waktu untuk $10^6$ percobaan per detik	Lama waktu untuk $10^{12}$ percobaan per detik
16 bit	$2^{16} = 65536$	32.7 milidetik	0.0327 mikrodetik
32 bit	$2^{32} = 4.3 \times 10^9$	35.8 menit	2.15 milidetik
56 bit	$2^{56} = 7.2 \times 10^{16}$	1142 tahun	10.01 jam
128 bit	$2^{128} = 4.3 \times 10^{38}$	$5.4 \times 10^{24}$ tahun	$5.4 \times 10^{18}$ tahun

Solusi: Kriptografer harus membuat kunci yang panjang dan tidak mudah ditebak.

## 2. Analytical attack

- Menganalisis kelemahan cipher secara matematik untuk menemukan parameter kunci, atau untuk mengurangi kemungkinan kunci yang tidak mungkin ada.
- Caranya: memecahkan persamaan-persamaan matematika (yang diperoleh dari konsep yang digunakan di dalam ciphernya) yang mengandung peubah-peubah yang merepresentasikan plainteks atau kunci.

- Contoh: Lihat kembali *Affine Cipher*  Enkripsi:  $C \equiv mP + b \pmod{n}$   
Dekripsi:  $P \equiv m^{-1}(C - b) \pmod{n}$   
Kunci:  $m$  dan  $b$

$m$  bilangan bulat yang relatif prima dengan  $n$

$b$  adalah jumlah pergeseran

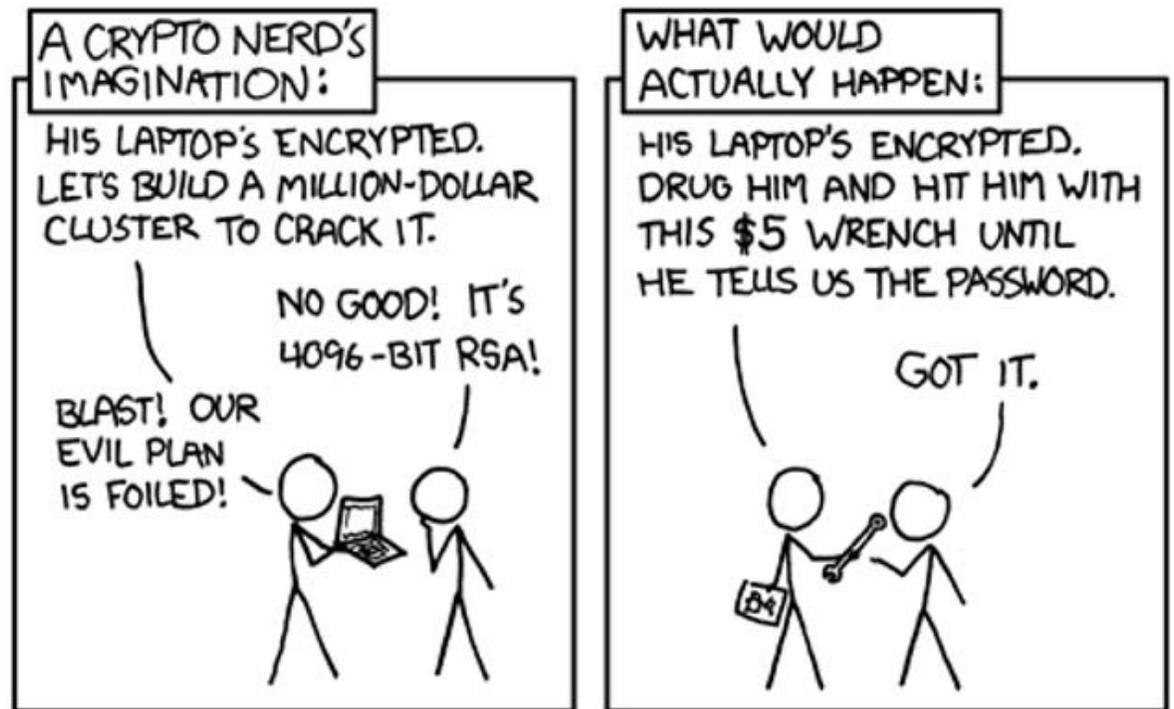
$m$  dan  $b$  dapat ditemukan dengan memecahkan dua buah persamaan linier yang memuat peubah  $m$  dan  $b$ , asalkan diketahui dua pasang plainteks dan cipherteks yang berkoresponden.

- Metode *analytical attack* biasanya lebih cepat menemukan kunci dibandingkan dengan *exhaustive attack*.
- Solusi: kriptografer harus membuat algoritma kriptografi yang sekompleks mungkin sehingga lebih sukar dianalisis

# Jenis-jenis Serangan

- Berdasarkan ketersediaan data yang digunakan untuk menyerang sistem kriptografi:

1. *Chipertext-only attack*
2. *Known-plaintext attack*
3. *Chosen-plaintext attack*
4. *Adaptive-chosen-plaintext attack*
5. *Chosen-chipertext attack*



## 1. *Ciphertext-only attack*

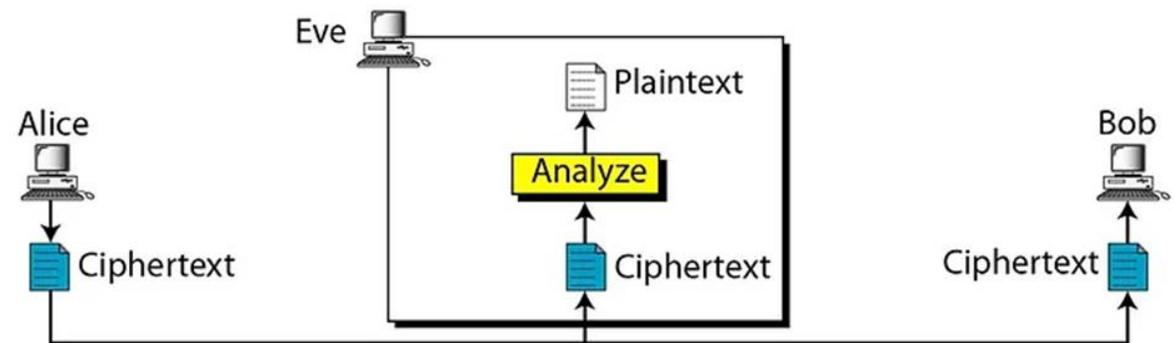
Kriptanalis hanya memiliki cipherteks saja. Ini serangan paling sulit karena informasi yang diperoleh hanya sedikit (cipherteks saja).

Teknik yang digunakan: *exhaustive key search*, terkaan, *metode* analisis frekuensi, dsb.

Diberikan:  $C_1 = E_k(P_1)$ ,  $C_2 = E_k(P_2)$ , ...,  $C_i = E_k(P_i)$

Deduksi:  $P_1, P_2, \dots, P_i$  atau  $k$  untuk mendapatkan  $P_{i+1}$  dari  $C_{i+1} = E_k(P_{i+1})$ .

## Ciphertext-Only Attack



hQIMAw3Jn/nLK/38ARAAAsSXLdhCtzUYKMptNxZImJXwhhIRm3QxfuyHjJ93ASylE  
e+6ABkuyFLJhiKryxp/JmS/alMPfF7hx2aTgovagaPzTwTV1jo6If2mhdCl6keed  
1Iz7C0f6jHIqq9d8g0bWDyvELEipn5LNDTX3Xp2Csx5ojRB2wckrUt1l1Xyj8G0H  
4DQUYbINRmJVu1JJC/acGvgOze66pHuRgSCxxHDscefjXenh/XejSYTo7aMi+Es7  
DCcD49zH6ZLDQN6B1N9q2oFI8QIhQ2y1QJbat1dWi/4yYWlKZcLKRSm8eo/gNCdL  
h9MncXBBSfgbvbu67CDZ9G05geZOn3LzQOpJ8hrZq/6K/uMcUKeZjW3RCo0T754f  
E5zYe1wUgtwS/lmQ2w5PQF/89bpshtDSYuL1fZgzrsE6DwophuCri5zwCGbEKlsI  
g6REIETFbZ2aCL4N2pZVunCIEuoP0zgEB6+M9egdpyxMsMqEBVg3AH7SalAtEguP  
T/MCxi0bZHCUhPupEKT8slbSrDNxTWMUXQt3XpL0bGCCrDMKLSowYfDiNnRkFbWK  
iiqw9hx4Q9CJg7xX7JRnVgwOeREiFnMYSbFlvPSxEou6FdBYhdqSefKin4Wnkmdw  
qrS18fjIW/kZ2v72uz0buEKkY9ubBox76yj1Ro9KUQMs3em03kc64959gTDiZ0qF  
AgwDrosDPQ2BeYQBD/9H5VKFw0an5j5MX1JpOSBAqNGKWq2bcEFnwJfk0DDlhyHD  
owHiG7gDowCS+5y/pf56v36HkzpJZATKqoRyKVxmQOxU9l3YnPc5fw8iFhxlrfcG  
ywzkJh/BRDQ/uy5fhGc/PbSm6iLv/SkkWTK8PSUD+g1yZyK0W7WkMh9QYS2OE7lQ  
qbwPNiy57reWkUWCoE4QmKqqpe7NXXM0eLT9l2D0hG2lthyvTvspkpxszl8+HMJv  
M2LMcY2FmmZWAJSdxsQSq9NQdyvCJX2D8oa89WQyXmp7mPXL7BQfoQNPndmn6Obi  
0EQojoemRNh14XNhMjPjxW7m34rH2gtvdN3Dg8iFrtocoVJqXqU3N+9T2sNe/bS8

## Cipherteks

- Contoh pada Caesar Cipher, penyerang hanya memiliki cipherteks berikut:

KHOORZRUOG

Penyerang tidak tahu kunci (pergeseran berapa huruf), hanya punya ciphertext itu saja.

Namun karena Caesar cipher hanya punya 25 kemungkinan pergeseran huruf, penyerang bisa mencoba semua pergeseran (brute force).

Saat dicoba pergeseran 3 huruf ke kiri, diperoleh plaintext:

HELLOWORLD

Serangan berhasil!

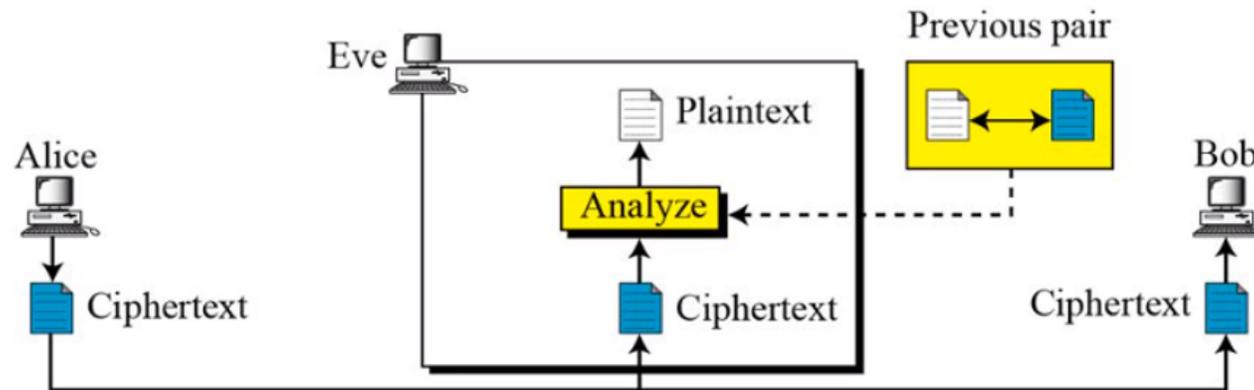
## 2. *Known-plaintext attack*

Diberikan sejumlah pasangan plainteks dan cipherteks yang berkoresponden:

$$P_1 \text{ dan } C_1 = E_k(P_1), \quad P_2 \text{ dan } C_2 = E_k(P_2), \quad \dots, \dots, \quad P_i \text{ dan } C_i = E_k(P_i)$$

Deduksi:  $k$  untuk mendapatkan  $P_{i+1}$  dari  $C_{i+1} = E_k(P_{i+1})$ .

## Known-Plaintext Attack



- Ini sering terjadi jika sebagian isi pesan bisa ditebak (misalnya header file, protokol standar, salam pembuka email).
- Beberapa pesan yang formatnya terstruktur membuka peluang untuk menerka plainteks dari cipherteks yang bersesuaian.

Contoh:

*From dan To* di dalam *e-mail*,

”Dengan hormat”, *wassalam*, pada surat resmi.

*#include, program*, di dalam *source code*

- Dengan menggunakan pasangan plainteks dan cipherteks, kriptanalis dapat menemukan kunci enkripsi (lihat pembahasan *cipher* klasik *affine cipher* dan *hill cipher*)

Dengan hormat

TF JOXUPOUXYT TRDSXQMONIYPEUFJDQUBGIMOCJQTNBEHCZEKROV  
BNTWLMVXMOWZLUCHOXYGSKBQGUAOBQZKIXYJIETSWVXHVKCUAOT  
OFYIZAKJGXKAWGQTRVFDZAJNQDUIWZCMYWNFIUPYMCZXIAKYUCQ  
IAZPIQMGAMGUAKKKHMWKDUXQDUAAKYOWEHLJPWYFKXSARBL LHGA  
JKTQNT RTPWSCIZASCGSLKVDHTUZSWBNBTJGYYUPQMFSYZAUTOQC  
DNGQMF SRLRTUWEMKADIVYLTJKFHLKJUWTSSHMHJFGTRIBYIDAHQ  
EPMPIQCROWDYRYZNSPNOJHQVKKTOCBPNFAJNLYJZNVBAYJWRGMC  
HJPWBDHHTPOXSIJVQWDMSIGMTRVEVXDILKVAYTNUNJXEZLAPGYE  
TRVZNVHSVWLGICDXQFOALDVPASUSYXPFHUWTILUQHTJQVGWFSPA  
EKBRBNIINYKHNTNUKJVDHVLXQKUZNVQXUOZZOJZYNPIVYSVFVTZ  
MMUUPWTGHRIOWCBKZYAGUMRCKHIQZSIGISPGBXPYXMOAWGAGHQV  
UWTEIGPBMOMBWIO PQEVKMRQATNBMI LHHLVUXGMOUWTZCLBKGWIJ  
HFRNGOSCMUHDWHBB

wassalam

- Contoh di dalam Caesar Cipher, misalkan penyerang tahu ciphertext:

KHOOR ZRUOG

dan juga tahu plaintext-nya seharusnya mengandung kata:

HELLO

Dari pasangan  $H \rightarrow K$ , penyerang tahu pergeseran  $(k) = 3$ .

Dengan kunci itu, semua ciphertext lain bisa didekripsi.

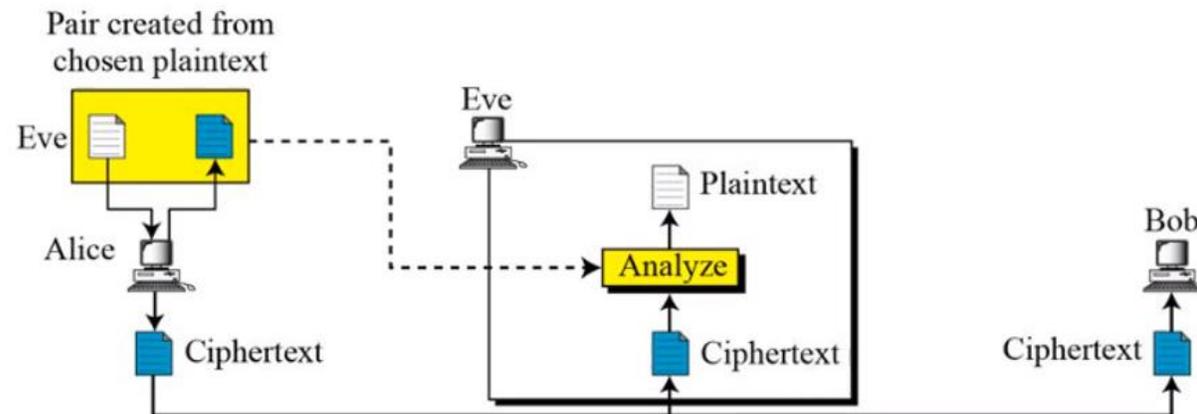
### 3. *Chosen-plaintext attack*

Kriptanalis dapat memilih plainteks sesuka hati untuk dienkrapsikan, yaitu plainteks-plainteks yang lebih mengarahkan penemuan kunci, lalu mengirim plainteks ke sistem enkripsi. Asumsi: Penyerang memiliki akses ke sistem enkripsi

Diberikan:  $P_1, C_1 = E_k(P_1), P_2, C_2 = E_k(P_2), \dots, P_i, C_i = E_k(P_i)$   
di mana kriptanalis dapat memilih diantara  $P_1, P_2, \dots, P_i$

Deduksi:  $k$  untuk mendapatkan  $P_{i+1}$  dari  $C_{i+1} = E_k(P_{i+1})$ .

## Chosen-Plaintext Attack



- Contoh pada Caesar Cipher, penyerang memilih plaintext sederhana, misalnya:

AAAAA

lalu mengirim plaintext ke sistem enkripsi

Sistem mengembalikan ciphertext, misalnya:

FFFFF

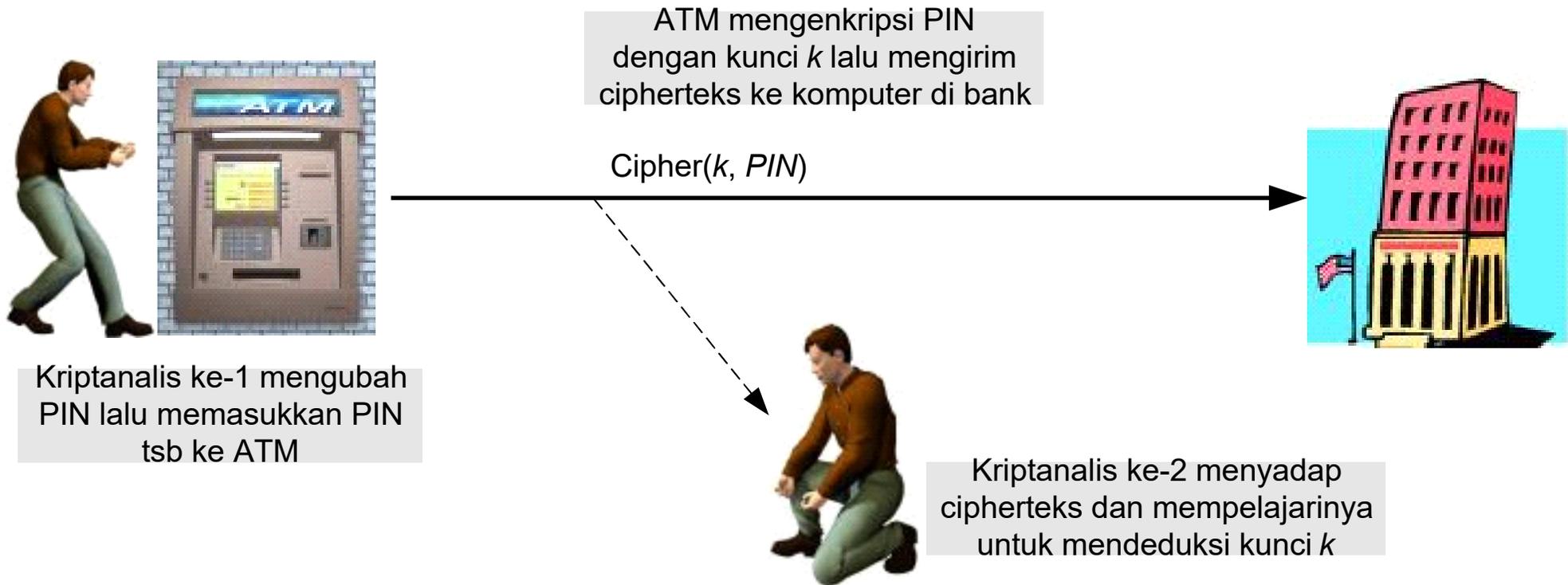
Dari sini terlihat huruf A  $\rightarrow$  F, berarti pergeseran = 5.

Dengan kunci itu, penyerang bisa mendekripsi semua ciphertext lain.

- Contoh di dalam Vigenere Cipher, ciphertext setiap huruf plainteks dihitung dengan:

$$C = (P + K) \text{ mod } 26$$

- Penyerang memilih plaintext  $P = \text{AAAAAAA...}$  (= 000000...).
- Maka ciphertext yang keluar,  $C$ , pasti sama dengan  $K$  ( $C = K$ ).
- Dengan mengetahui  $K$ , ciphertext lain bisa langsung dicoba didekripsi dengan  $K$  tersebut.



*Chosen-plaintext attack*

## 4. *Adaptive-chosen-plaintext attack*

- *Adaptive chosen plaintext attack* adalah variasi dari *chosen plaintext attack*, dalam hal ini penyerang **tidak hanya memilih plaintext**, tapi dapat **menyesuaikan plaintext berikutnya** berdasarkan hasil *ciphertext* dari *query* sebelumnya.
- Artinya, penyerang belajar secara bertahap: setelah melihat hasil enkripsi pertama, ia memilih *plaintext* baru yang lebih “cerdas” untuk menggali informasi lebih dalam tentang algoritma/kunci.

- Contoh di dalam Caesar Cipher, penyerang memilih plaintext:

A

→ Ciphertext keluar: F. Artinya pergeseran mungkin 5.

Untuk memastikan, penyerang pilih plaintext lain:

B

→ Ciphertext keluar: G. Polanya konsisten, maka kunci pergeseran dipastikan = 5.

- Contoh di dalam Vigenere Cipher

- Enkripsi setiap huruf plainteks:

$$C = (P + K) \text{ mod } 26$$

- Penyerang adaptif:

- Pertama masukkan plaintext = AAAA (=0000), maka ciphertext C = K.
- Setelah tahu K, ia coba plaintext lain untuk memverifikasi, misalnya BBBB (=1111), hasilnya menegaskan hasil enkripsi dengan K yang sama.
- Dengan informasi itu, ciphertext lain bisa langsung dipecahkan.

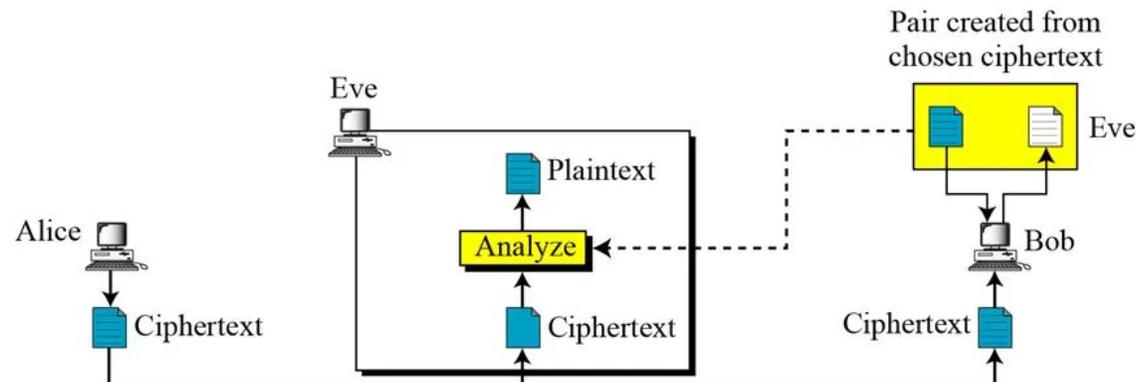
## 5. Chosen-ciphertext attack

Penyerang dapat memilih ciphertext tertentu dan mendekripsinya dengan oracle yang bisa mendekripsi, dan menggunakan hasil dekripsi untuk memecahkan cipher atau menemukan kunci.

$$C_1, P_1 = D_k(C_1), C_2, P_2 = D_k(C_2), \dots, C_i, P_i = D_k(C_i)$$

Deduksi:  $k$  (yang mungkin diperlukan untuk mendekripsi pesan pada waktu yang akan datang).

### Chosen-Ciphertext Attack



- Contoh di dalam Caesar cipher, penyerang memilih ciphertext:

$C = 'A'$

lalu sistem dekripsi menghasilkan plaintext:

$P = 'X'$

berarti penyerang mengetahui pergeseran huruf  $(k) = 3$

Sebuah algoritma kriptografi dikatakan aman secara komputasi (*computationally secure*) bila ia memenuhi tiga kriteria berikut:

1. Persamaan matematika yang menggambarkan operasi di dalam algoritma kriptografi sangat kompleks sehingga algoritma tidak mungkin dipecahkan secara analitik.
2. Biaya untuk memecahkan cipherteks melampaui nilai informasi yang terkandung di dalam cipherteks tersebut.
3. Waktu yang diperlukan untuk memecahkan cipherteks melampaui lamanya waktu informasi tersebut harus dijaga kerahasiaannya.