

# PHÁT HIỆN LỖI VÀ LỖ HỔNG PHẦN MỀM

Lỗi hỏng phần mềm

1

Lỗi hỏng do lập trình

2

Lỗi hỏng sử dụng mật mã

3

Lỗi hỏng mạng

4

Lỗi hỏng web

1

Lỗi hỏng do lập trình

2

Lỗi hỏng sử dụng mật mã

3

Lỗi hỏng mạng

4

Lỗi hỏng web

## **Buffer Overruns**

**là những lỗi hỏng do sai sót trong thao tác với bộ  
nhớ đệm**

# Buffer Overruns

## ❑ Ví dụ 1

```
void echo() {  
    char msg[1024];  
    gets(msg);  
    puts(msg);  
}
```

CWE-121: Stack-based Buffer Overflow

## ❑ Ví dụ 2

```
int get_user(char* user)  
{  
    char buf[16];  
    if(strlen(user) <= sizeof(buf))  
        strcpy(buf, user);  
}
```

CWE-193: Off-by-one Error

# Buffer Overruns

- **Ngôn ngữ bị ảnh hưởng:**
  - Assembly
  - C
  - C++ (nếu chỉ sử dụng phần OOP thì ít hơn)
- **Hậu quả:** rất khác nhau, từ DoS đến thực thi mã tùy ý

# Buffer Overruns

---

- Giải pháp ngăn ngừa
  - Replace Dangerous String Handling Functions
  - Check Loops and Array Accesses
  - Replace C String Buffers with C++ Strings
  - Replace Static Arrays with STL Containers
  - Use Analysis Tools

# Buffer Overruns

## ❑ Dangerous String Handling Functions

```
char *gets(char *string)
```

```
char *strcpy( char *dest, const char *src )
```

...

## ❑ Safe alternatives

```
char *fgets(char *string, int value, FILE *stream)
```

```
error_t strcpy_s(char *dest, rsize_t destsz, const char *src)
```

...



## **Format String**

**là những lỗ hổng do sai sót trong sử dụng các hàm xử lý chuỗi có tham số định dạng**

# Format String

## ❑ Tương ứng với CWE

- CWE-134: Uncontrolled Format String

## ❑ Ví dụ

```
void printWrapper(char *string) {  
    printf(string);  
}
```

```
int main(int argc, char **argv) {  
    printWrapper(argv[1]);  
    return (0);  
}
```

Dữ liệu đầu vào (user input) được sử dụng như format string

# Format String

- Ngôn ngữ bị ảnh hưởng
  - C/C++ bị ảnh hưởng rõ ràng nhất; hậu quả có thể lên đến việc thực thi mã tùy ý
  - Các ngôn ngữ khác cũng có thể bị ảnh hưởng nhưng không dẫn tới việc thực thi mã tùy ý
- Ngăn ngừa
  - Không đưa trực tiếp dữ liệu người dùng vào format string
  - Với C/C++, luôn sử dụng lời gọi `printf("%s", user_input);`

## **Integer Overflows**

**là những lỗi hổng do chương trình thao tác trên  
biến nguyên với những giá trị vượt quá phạm vi  
biểu diễn**

# Integer Overflows

## ❑ Đoạn mã ví dụ từ OpenSSH 3.3

```
nresp = packet_get_int();  
if (nresp > 0)  
{  
    response = xmalloc(nresp*sizeof(char*));  
    for (i = 0; i < nresp; i++)  
        response[i] = packet_get_string(NULL);  
}
```

- sizeof(char\*) = 4
- Nếu nresp = 0x40000020 kết quả phép nhân là 0x80
- Cấp phát 0x80 bytes
- Nhưng đọc vào nresp chuỗi → tràn!

# Integer Overflows

## ☐ Sự tương ứng với các CWE

- CWE-128: Wrap-around Error
- CWE-682: Incorrect Calculation
- CWE-190: Integer Overflow or Wraparound
- CWE-191: Integer Underflow (Wrap or Wraparound)
- CWE-192: Integer Coercion Error

# Integer Overflows

- Ngôn ngữ bị ảnh hưởng
  - Mọi ngôn ngữ đều bị ảnh hưởng, nhưng ở mức độ khác nhau
  - Integer Overflow ở C/C++ có thể dẫn tới Buffer Overflow, và kéo theo việc thực thi mã tùy ý
  - Với các ngôn ngữ khác thì dừng ở DoS và các vấn đề logic

# Integer Overflows

- ❑ Giải pháp ngăn ngừa: thay thế phép toán có thể dẫn tới tràn, kiểm tra điều kiện tràn



## **Command Injection**

**là lỗi hổng xuất hiện khi dữ liệu người dùng được sử dụng để xây dựng câu lệnh truyền đến trình thông dịch**

# Command Injection

## ❑ Ví dụ

```
char buf[1024];
```

```
char cmd[2048];
```

```
fgets(buf, sizeof(buf), stdin);
```

```
sprintf(cmd, "echo %s", buf);
```

```
system(cmd);
```

```
root@kali:~# ./hello
Enter some thing: NetPro
NetPro
root@kali:~# ./hello
Enter some thing: NetPro; ls -l
NetPro
total 56
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Desktop
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Documents
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Downloads
-rw-r--r-- 1 root root 80 Dec 13 10:45 hash.txt
-rwxr-xr-x 1 root root 15620 Dec 14 05:20 hello
-rw-r--r-- 1 root root 197 Dec 14 05:20 main.c
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Music
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Pictures
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Public
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Templates
drwxr-xr-x 2 root root 4096 Nov 25 13:44 Videos
root@kali:~#
```

# Command Injection

## ❑ Sự tương ứng với các CWE

- CWE-77: Failure to Sanitize Data into a Control Plane

## ❑ Ngôn ngữ bị ảnh hưởng

Bất kỳ ngôn ngữ nào

## ❑ Ngăn ngừa

- Tránh việc phải sử dụng đến trình thông dịch
- Nếu vẫn dùng thì cần lọc dữ liệu

## **Race Conditions**

**là lỗi hổng xuất hiện do chương trình không nhận  
thấy điều kiện thực thi đã thay đổi**

# Race Conditions

- Race Condition = TOCTOU
  - TOCTTOU – Time of Check To Time of Use
  - Check – Kiểm tra điều kiện
  - Use – Thực thi tác vụ (nếu điều kiện được thỏa mãn)
- Lỗi hỏng xuất hiện trong môi trường mà có nhiều chủ thể (chương trình...) cùng thao tác trên một đối tượng. Một chủ thể làm thay đổi đối tượng mà chủ thể khác không kịp nhận biết sự thay đổi đó.

# Thông tin thêm. Kiểm soát truy cập

- Access control: người dùng chỉ được phép truy cập file khi có thẩm quyền tương ứng
- SEUID: thuộc tính cho phép chương trình chạy với quyền nào đó khác với quyền của người dùng
  - Một số chương trình (ping, mount...) được setuid root để có được thẩm quyền truy cập hệ thống, dù được chạy bởi user nào
- System call `access()` cho phép kiểm tra thẩm quyền của người dùng

# Thông tin thêm. Kiểm soát truy cập

Not secure pwnable.kr/play.php

Gmail YT NetHD FB VnE DanTri VNnet C24 C24 CTF Fuzzing Test S

[Toddler's Bottle]

fd - 1 pt [writeup]

Mommy! what is a file descriptor in Linux?

\* try to play the wargame your self but if you are ABSOLUTE beginner, follow this tutorial link:  
<https://youtu.be/971eZhMHQQw>

ssh fd@pwnable.kr -p2222 (pw:guest)

[fd] [collision]

# Thông tin thêm. Kiểm soát truy cập

```
fd@prowl:~$ ls -l
total 16
-rsr-x--- 1 fd_pwn fd 7322 Jun 11 2014 fd
-rw-r--r-- 1 root root 418 Jun 11 2014 fd.c
-r--r----- 1 fd_pwn root 50 Jun 11 2014 flag
fd@prowl:~$ cat flag
cat: flag: Permission denied
fd@prowl:~$
```



# Thông tin thêm. Kiểm soát truy cập

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char buf[32];
int main(int argc, char* argv[], char* envp[]){
    int fd = atoi( argv[1] ) - 0x1234;
    int len = 0;
    len = read(fd, buf, 32);
    if(!strcmp("LETMEWIN\n", buf))
        system("/bin/cat flag");
    else
        printf("learn about Linux file IO\n");
    return 0;
}
```

# Thông tin thêm. Kiểm soát truy cập

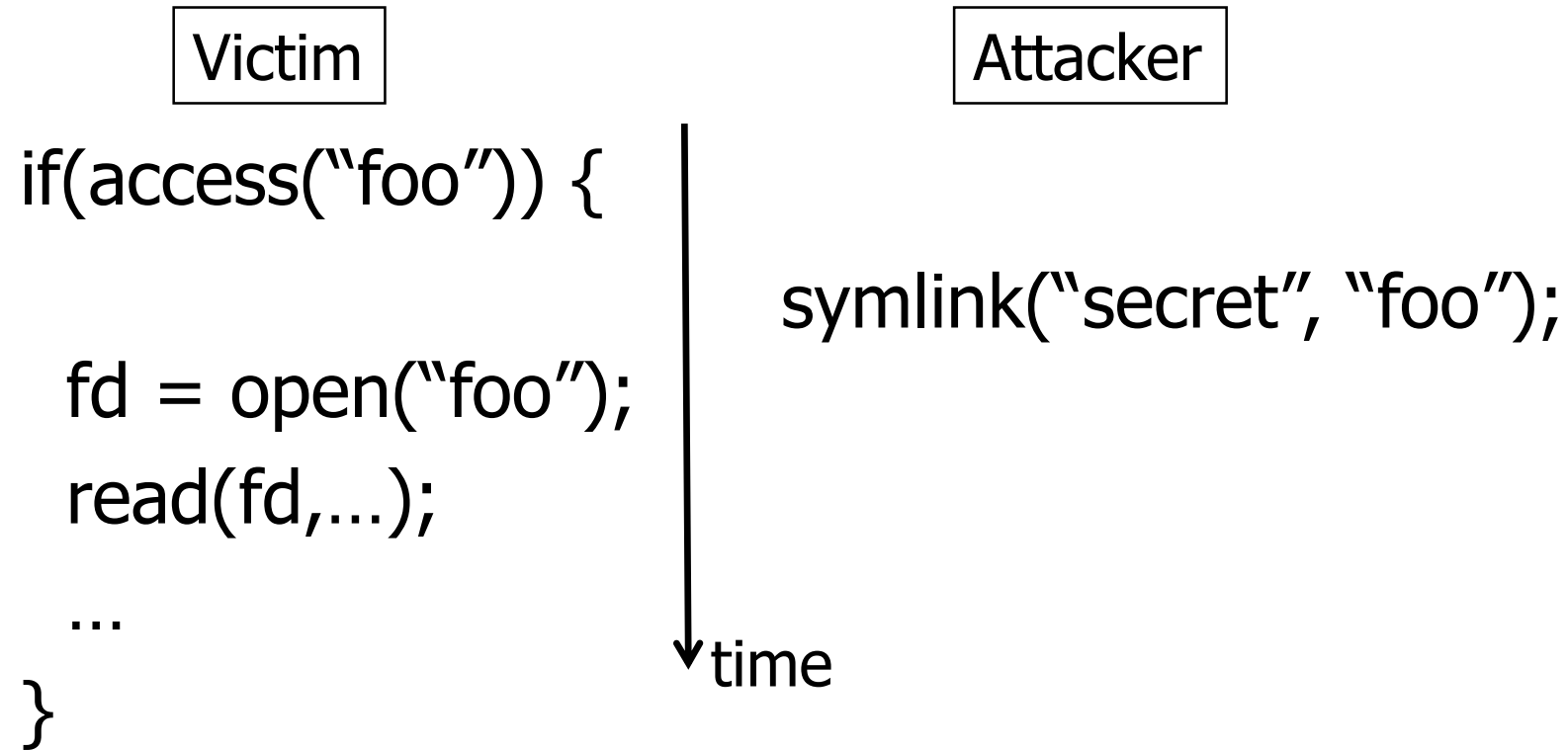
```
fd@prowl:~$ python -c 'print 0x1234'
4660
fd@prowl:~$ ./fd 4660
LETMEWIN
good job :)
mommy! I think I know what a file descriptor is!!
fd@prowl:~$
```

# Thông tin thêm. Kiểm soát truy cập

- Chương trình A: có năng lực lớn, có khả năng truy cập các đối tượng H, L
- Người dùng U: không được phép truy cập H, chỉ được truy cập L
- U lợi dụng A để truy cập H
  - U yêu cầu A truy cập L
  - A kiểm tra thẩm quyền → OK → chuẩn bị...
  - U xóa L cũ, tạo L mới là symlink của H
  - ...truy cập L!!!!

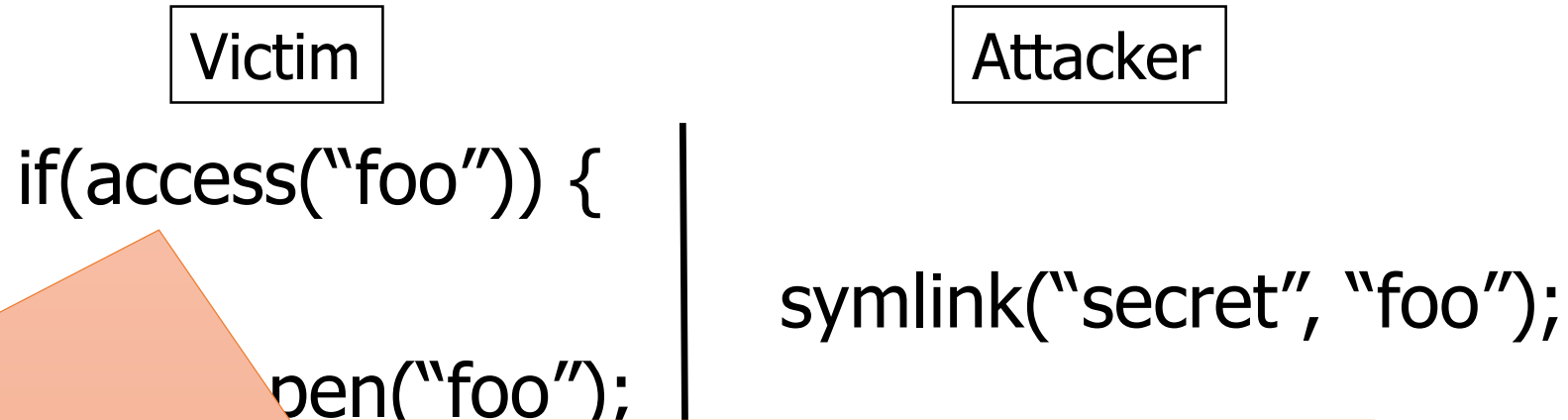
# Race Conditions. File System

- **Tấn công:** người dùng bình thường muốn đọc file vượt quá thẩm quyền.



# Race Conditions. File System

- **Tấn công:** người dùng bình thường muốn đọc file vượt quá thẩm quyền.

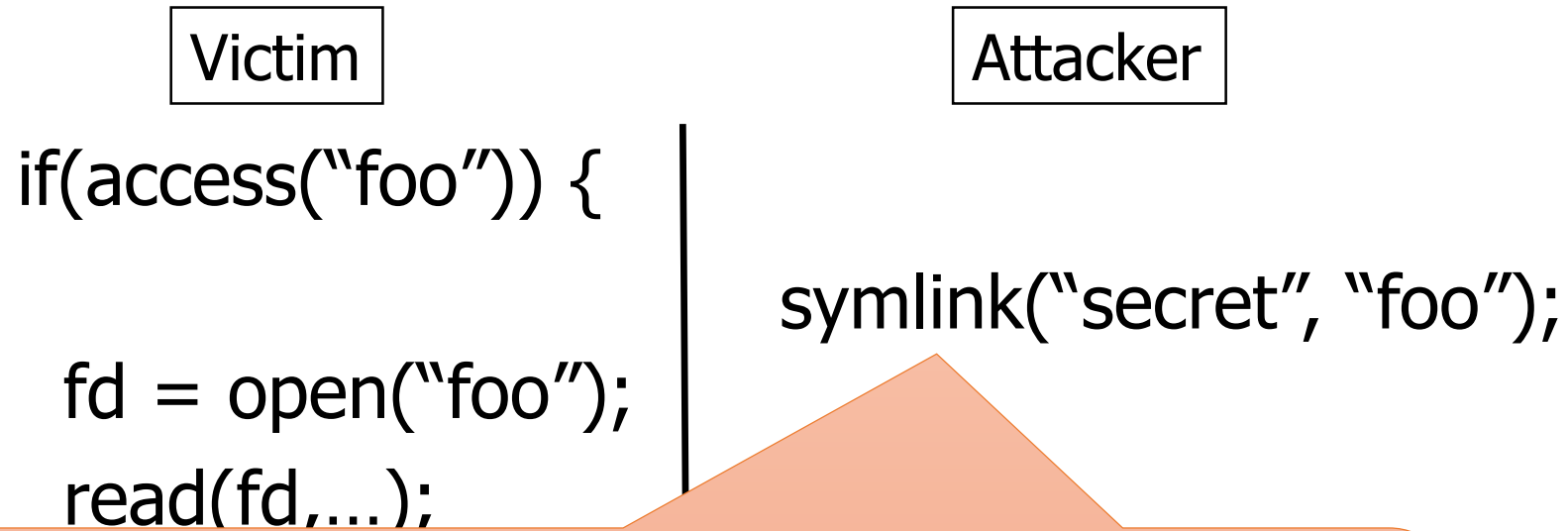


## Bước 1:

- Attacker tạo symlink "foo" tới một file bất kỳ mà attacker có quyền truy cập
- Attacker yêu cầu một chương trình setuid-root truy cập file "foo"
- Victim kiểm tra quyền truy cập của attacker. Kết quả là OK → chuẩn bị đáp ứng yêu cầu truy cập.

# Race Conditions. File System

- **Tấn công:** người dùng bình thường muốn đọc file vượt quá thẩm quyền.

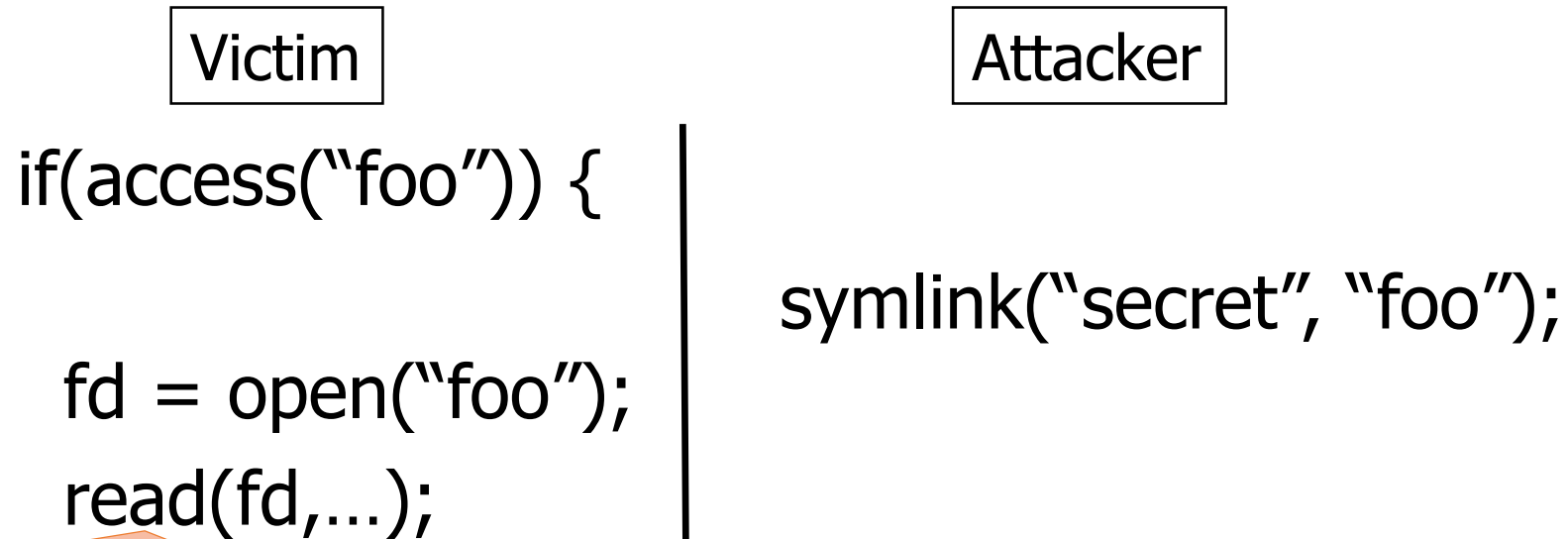


Bước 2:

- Ở thời điểm **sau hàm `access()` và trước hàm `open()`** của Victim, Attacker thay đổi symlink, trỏ đến file không có quyền truy cập "secret".

# Race Conditions. File System

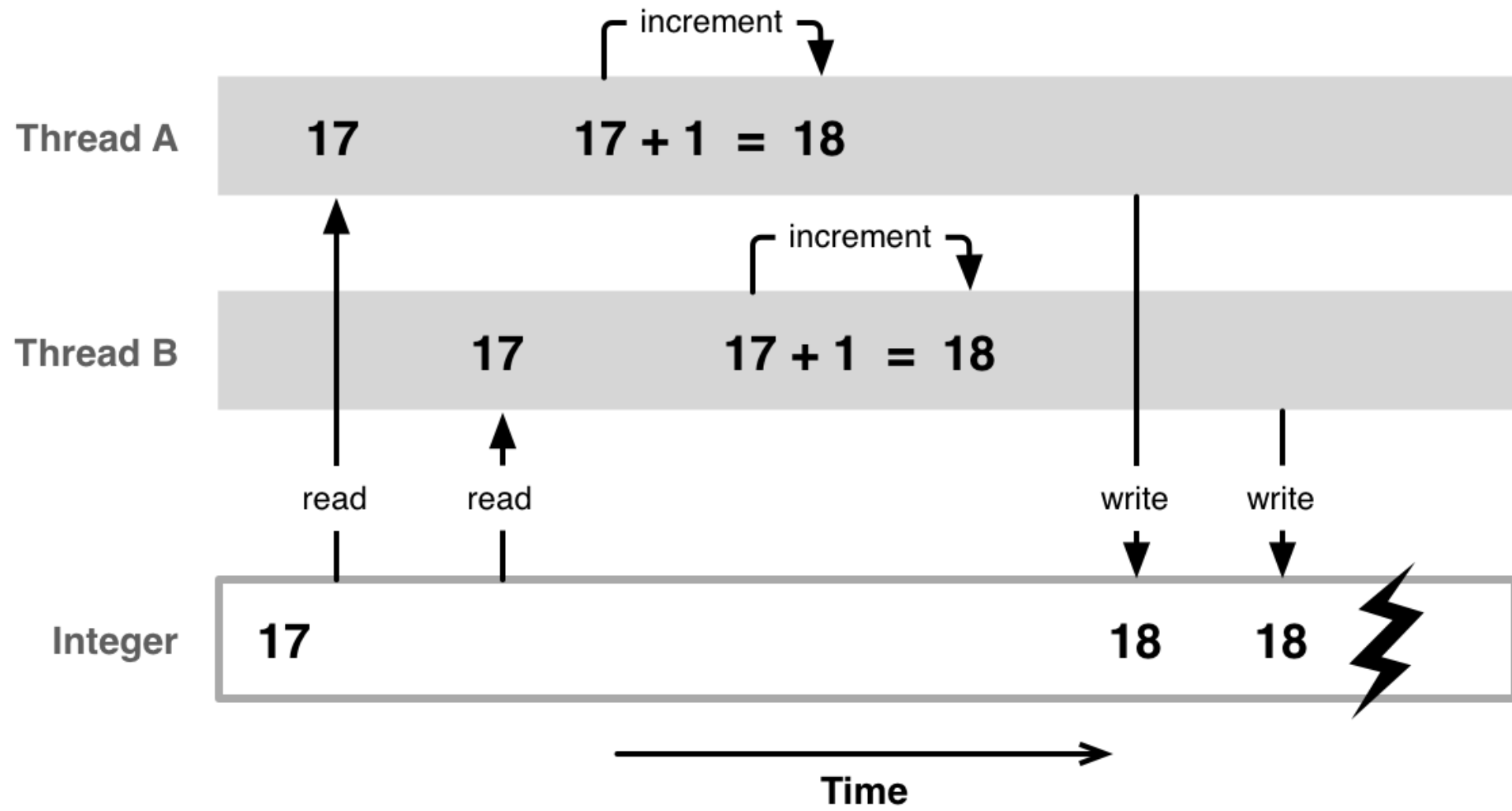
- **Tấn công:** người dùng bình thường muốn đọc file vượt quá thẩm quyền.



Bước 3:

- Victim thực hiện truy cập file "foo" khi mà nó không còn trỏ đến file ban đầu nữa.
- Hacker đạt mục đích!

# Race Condition. Threaded Data Access





# Race Conditions

## ☐ Sự tương ứng với các CWE

- CWE-362: Race Condition (parent)
- CWE-364: Signal Handler Race Condition
- CWE-365: Race Condition in Switch
- CWE-366: Race Condition Within a Thread
- CWE-367: Time-of-Check Time-of-Use (TOCTOU) Race Condition
- CWE-368: Context Switching Race Condition
- CWE-370: Race Condition in Checking for Certificate Revocation
- CWE-421: Race Condition During Access to Alternate Channel

# Race Conditions

---

## □ Ngôn ngữ bị ảnh hưởng

- Mọi ngôn ngữ đều bị ảnh hưởng

## □ Ngăn ngừa

- Nói chung rất phức tạp
- Sử dụng cơ chế đồng bộ, cơ chế khóa là một giải pháp

1

Lỗi hỏng do lập trình

2

Lỗi hỏng sử dụng mật mã

3

Lỗi hỏng mạng

4

Lỗi hỏng web

**Use of Weak Password-Based Systems** là để chỉ  
những lỗ hổng liên quan đến cơ chế sử dụng mật  
khẩu

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Password Aging
- CWE 263: Password Aging with Long Expiration
- CWE 521: Weak Password Requirements
- CWE 522: Insufficiently Protected Credentials
- CWE 620: Unverified Password Change
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Password Aging

- CWE 263: Passwords Stored in Plain Text
- CWE 521: Passwords Stored in Source Code
- CWE 522: Passwords Stored in Configuration Files
- CWE 620: Passwords Stored in Log Files
- CWE 549: Passwords Stored in Database
- CWE 640: Passwords Stored in Memory

Mật khẩu có thể bị khám phá bằng dịch ngược

```
...  
Properties prop = new Properties();  
prop.load(new FileInputStream("config.properties"));  
String password =  
Base64.decode(prop.getProperty("password"));  
DriverManager.getConnection(url, usr, password);  
...
```

ord

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Password Aging
- CWE 263: Passwords with Long Expiration
- CWE 521: Weak Password Storage
- CWE 522: Weak Password Complexity
- CWE 620: Weak Password Requirements
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

Mật khẩu được bảo vệ bằng hàm biến đổi không an toàn (base64, Michael64,...) có thể bị khám phá

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Password Aging
- CWE 263: Password Aging with Long Expiration
- CWE 521: Weak Password Requirements
- CWE 522: Passwords with Long Expiration
- CWE 625: Passwords with Long Expiration
- CWE 540: Passwords with Long Expiration
- CWE 640: Passwords with Long Expiration

Không giới hạn thời gian sử dụng mật khẩu, hoặc giới hạn quá lớn. Dùng càng lâu thì khả năng bị lộ, bị khám phá càng cao.

Password



# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Password Aging
- CWE 263: Password Aging with Long Expiration
- CWE 521: Weak Password Requirements
- CWE 522: Insufficiently Protected Credentials
- CWE 620: Improperly Implemented Password Change
- CWE 549: Password Comparison Error
- CWE 640: Password Policy Not Enforced

Mật khẩu đơn giản có thể dễ dàng bị dò đoán bằng tấn công từ điển

Password

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using Passwords
- CWE 263: Password Age
- CWE 521: Weak Password
- CWE 522: Insufficiently
- CWE 620: Unverified Password Change
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

"Câu hỏi riêng tư"  
Cơ chế khôi phục yếu thì tài khoản có khả năng  
cao bị chiếm đoạt

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptography for Passwords
- CWE 262: Not Using F
- CWE 263: Password A
- CWE 521: Weak Pass
- CWE 522: Insufficiently Protected Credentials
- CWE 620: Unverified Password Change
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

Hiện dấu '\*' khi nhập mật khẩu nhằm mục đích che giấu, tránh bị xem trộm

# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

- CWE 259: Hard-Coded Password
- CWE 261: Weak Cryptographic Key
- CWE 262: Not Using Secure Random Number Generation
- CWE 263: Password Stored in Plain Text
- CWE 521: Weak Password
- CWE 522: Insufficiently Protected Credentials
- CWE 620: Unverified Password Change
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

Không có cơ chế xác nhận đối với yêu cầu đổi mật khẩu → tài khoản có thể bị chiếm đoạt bởi attacker

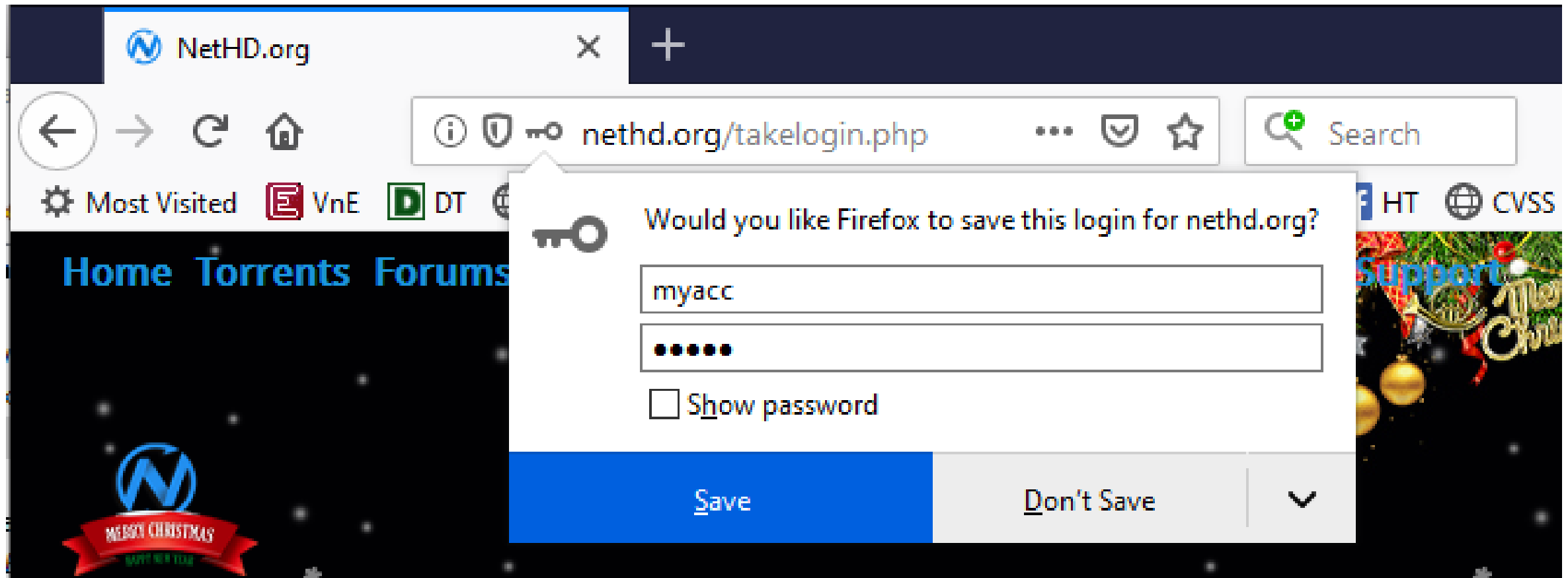
# Use of Weak Password-Based Systems

## ❑ Các CWE liên quan đến password

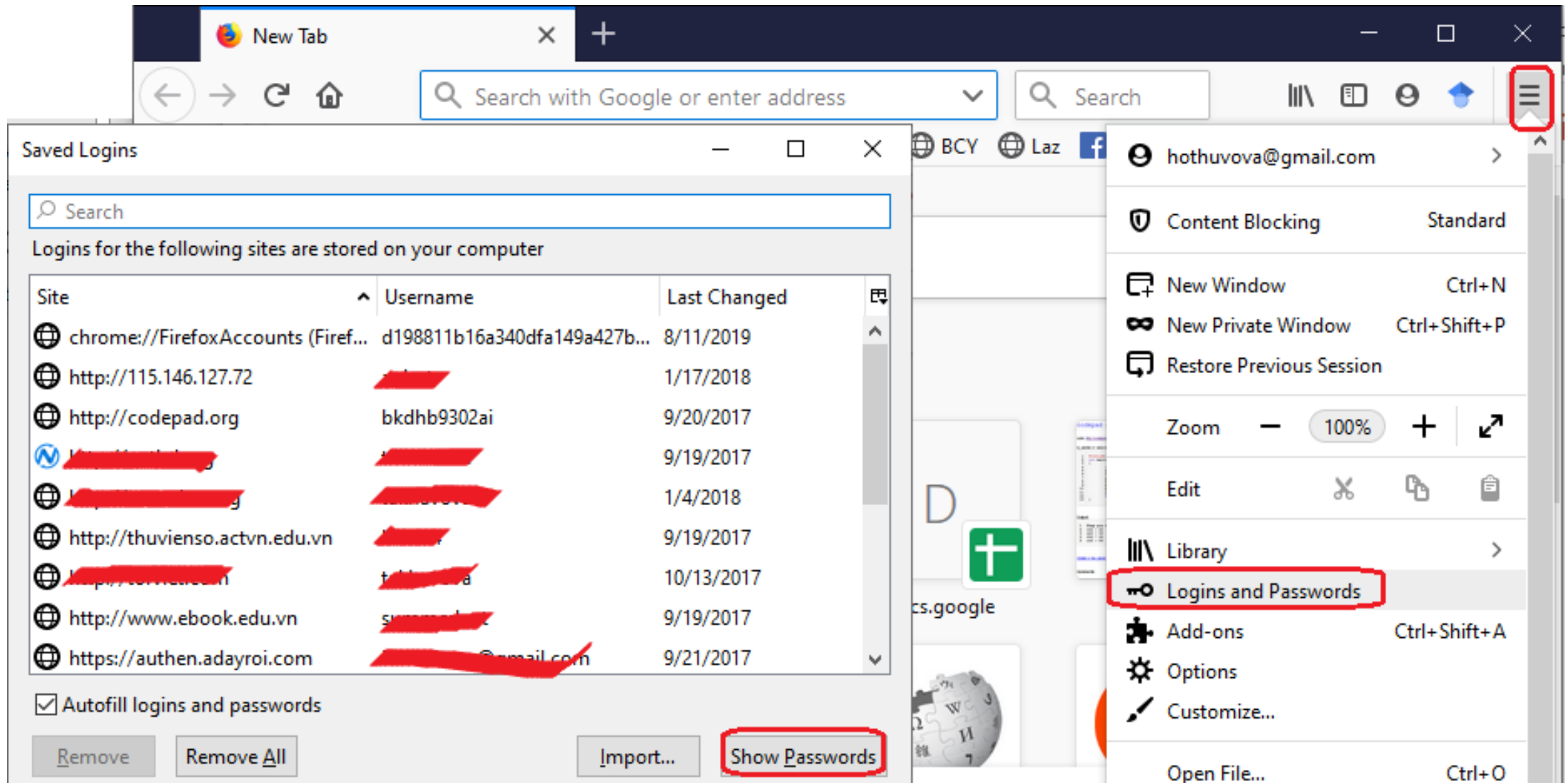
- CWE 259: Hard-Coded Credentials
- CWE 261: Weak Cryptographic Keys
- CWE 262: Not Using Secure Storage
- CWE 263: Password Aging with Long Expiration
- CWE 521: Weak Password Requirements
- CWE 522: Insufficiently Protected Credentials
- CWE 620: Unverified Password Change
- CWE 549: Missing Password Field Masking
- CWE 640: Weak Password Recovery Mechanism for Forgotten Password

Cơ chế lưu trữ hoặc truyền tải mật khẩu không đảm bảo an toàn

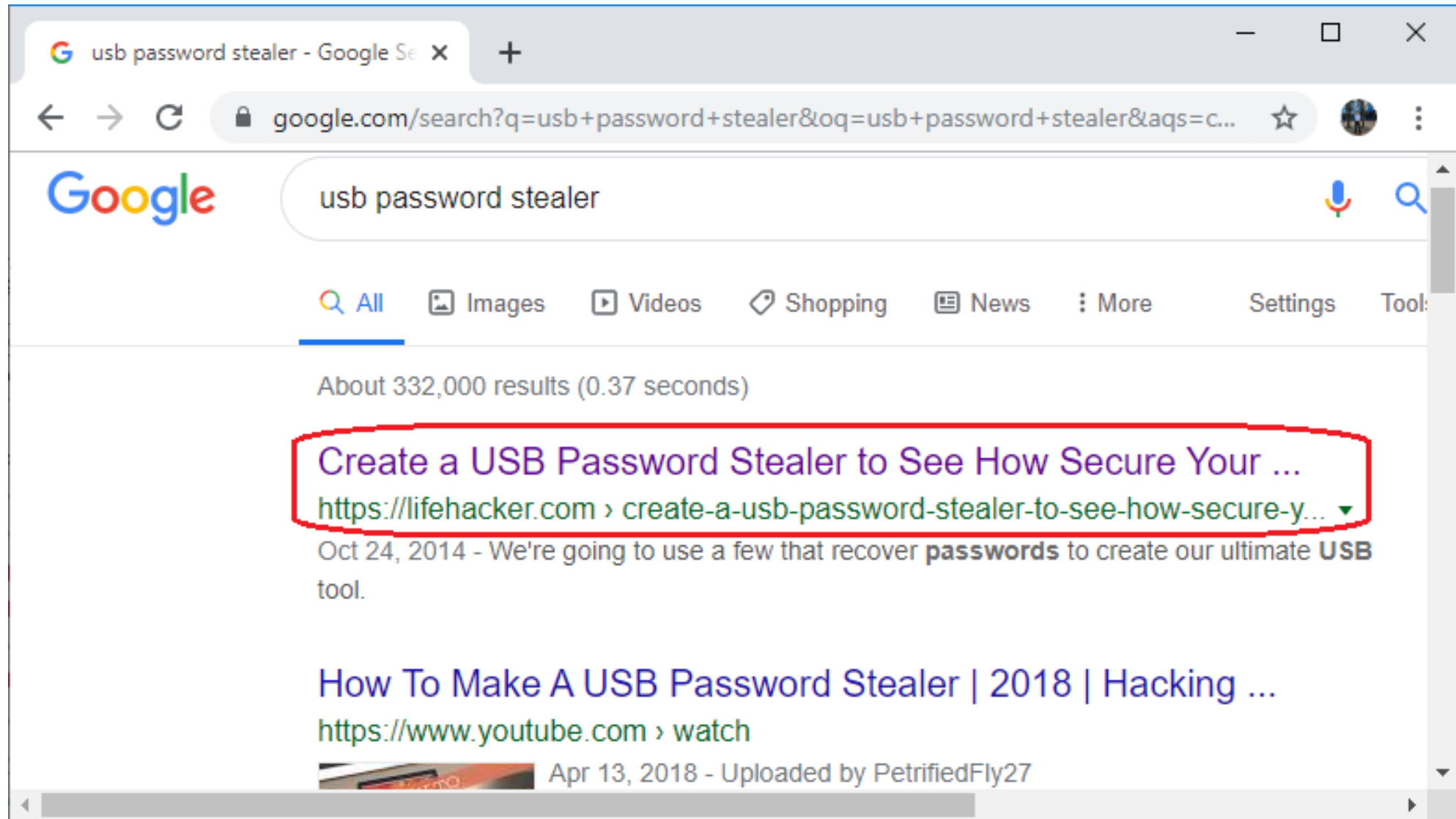
# Lưu trữ mật khẩu bởi trình duyệt



# Lưu trữ mật khẩu bởi trình duyệt



# Lưu trữ mật khẩu bởi trình duyệt





# Lưu trữ mật khẩu bởi trình duyệt

Secure | [nirsoft.net/utils/index.html#password\\_utils](http://nirsoft.net/utils/index.html#password_utils)

## IE PassView v1.42

IE PassView is a small utility that reveals the passwords stored by Internet Explorer 11.0, as well as older versions of Internet Explorer, v4.0 - v6.0

## PasswordFox v1.60

PasswordFox is a small password recovery tool that allows you to view the passwords stored in your current Firefox profile. By default, PasswordFox displays the passwords stored in your current Firefox profile. For each password entry, the following information is displayed: Password, User Name Field, Password Field, and the Signons filename.

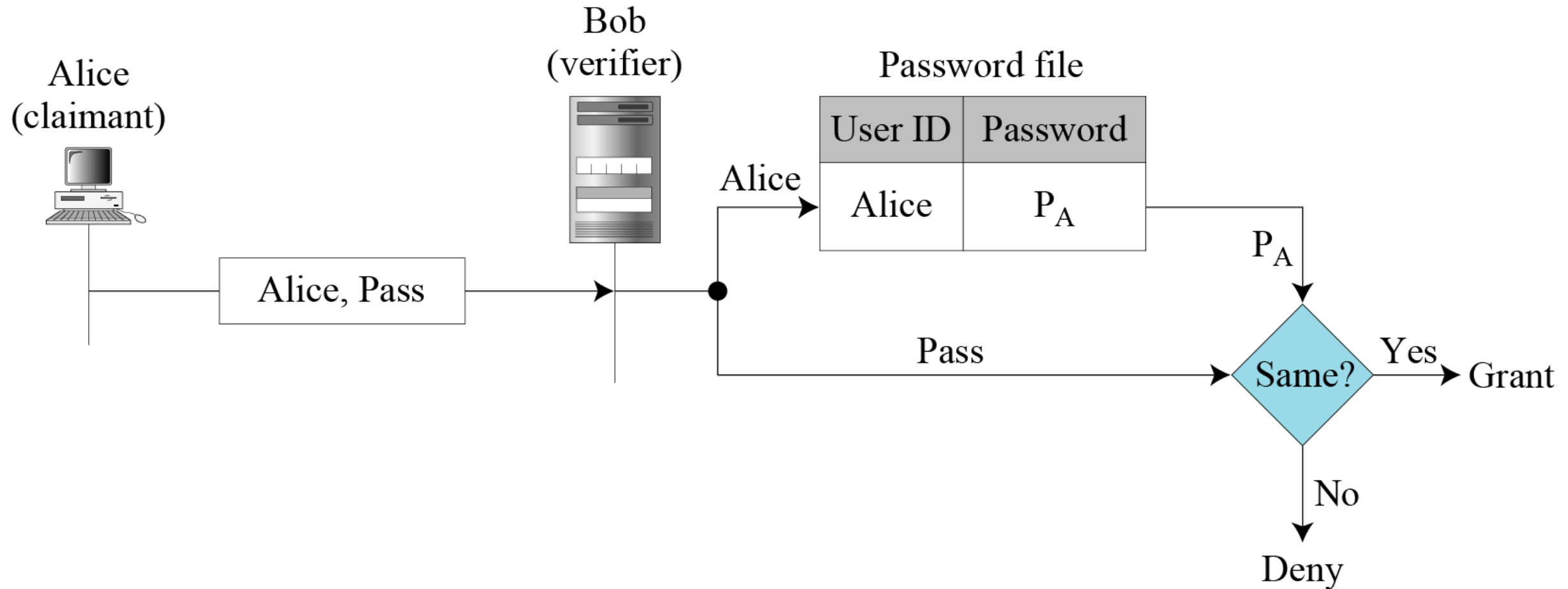
## ChromePass v1.46

ChromePass is a small password recovery tool that allows you to view the passwords stored in your current Chrome browser. For each password entry, the following information is displayed: User Name, Password, and Created Time. You can select one or more items

# Lưu mật khẩu ở máy chủ. Dạng rõ

$P_A$ : Alice's stored password

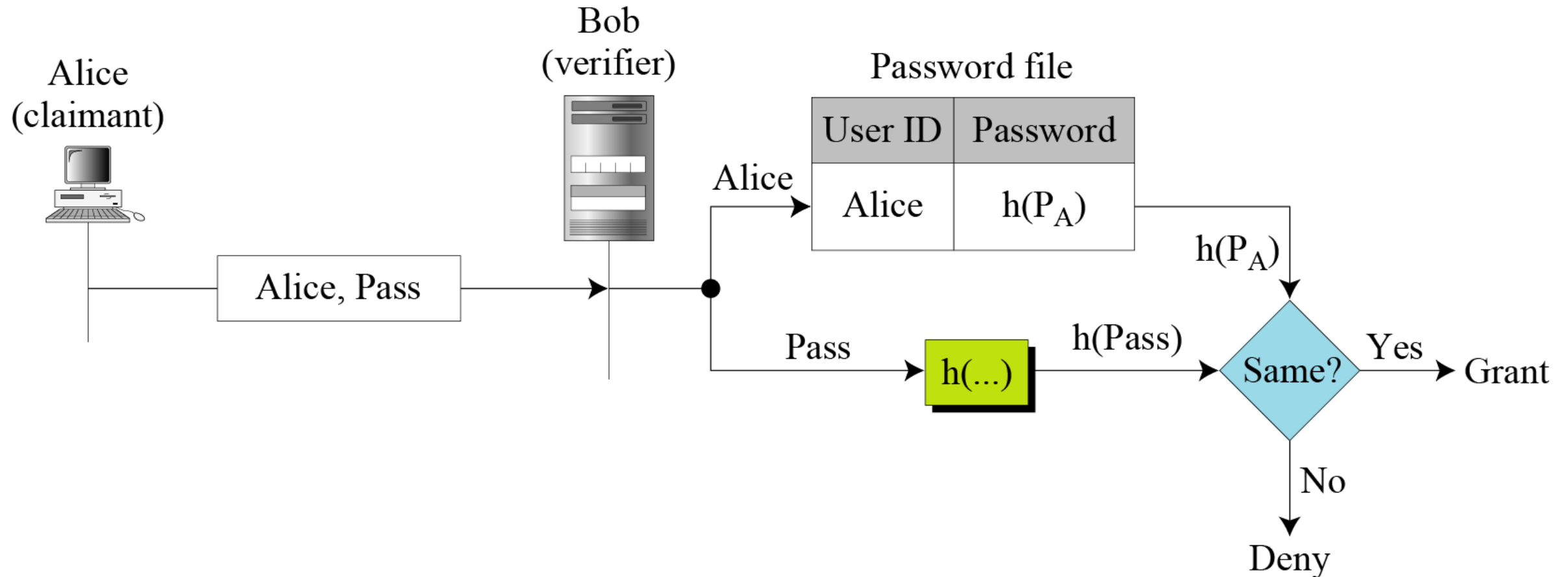
Pass: Password sent by claimant



# Lưu mật khẩu ở máy chủ. Dạng băm

$P_A$ : Alice's stored password

Pass: Password sent by claimant

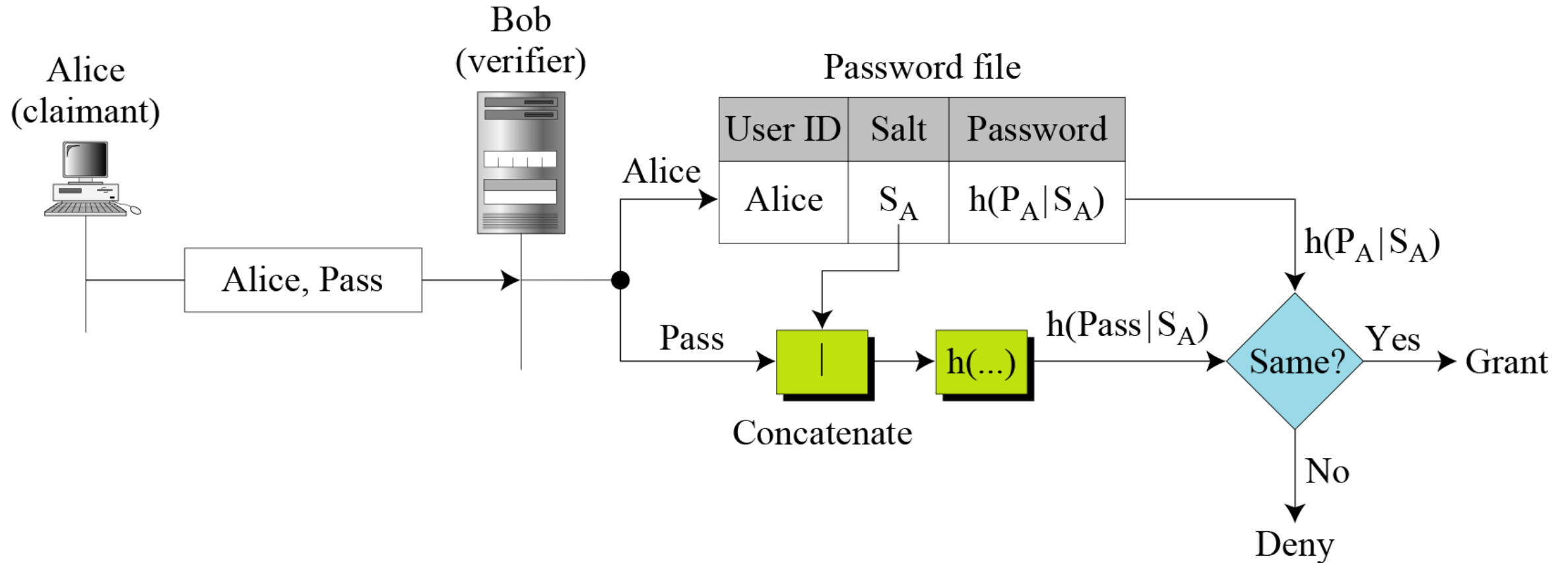


# Lưu mật khẩu ở máy chủ. Dạng băm có salt

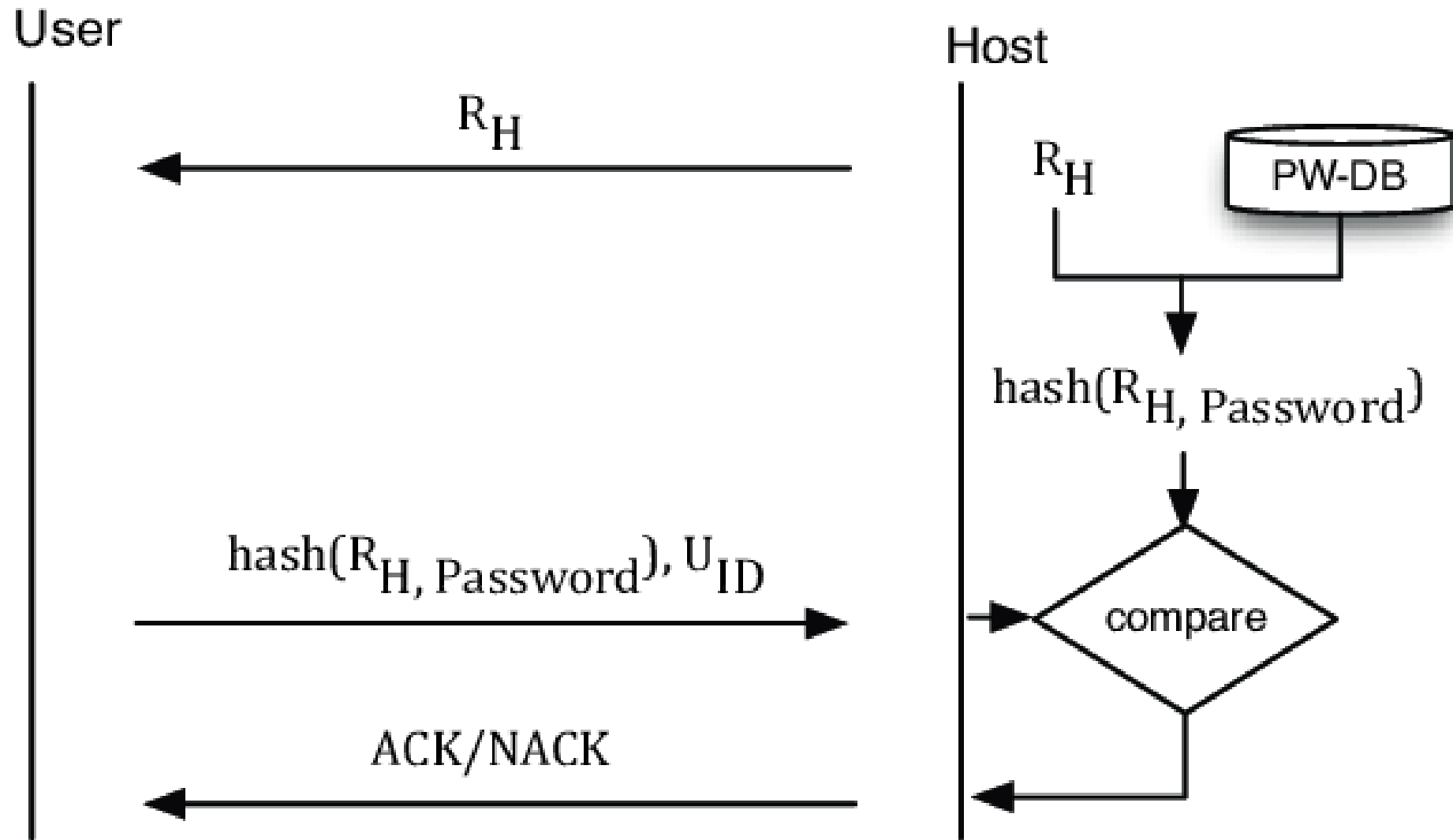
$P_A$ : Alice's password

$S_A$ : Alice's salt

Pass: Password sent by claimant



# Lưu mật khẩu ở dạng rõ nhưng bảo vệ khi truyền



# Quản lý mật khẩu an toàn

- Lựa chọn mô hình lưu trữ mật khẩu
- Quy định độ phức tạp của mật khẩu
- Quy định thời hạn sử dụng mật khẩu và buộc phải đổi mật khẩu khi hết hạn
- Quy định thời hạn được phép sử dụng lại mật khẩu
- Thiết lập cơ chế chống dò mật khẩu

**Weak Random Numbers** chỉ những lỗi hổng  
xuất hiện do sử dụng số "ngẫu nhiên"  
không an toàn

# Số ngẫu nhiên

- Số ngẫu nhiên là những số có xác suất phân phối đều, và không thể dự đoán được giá trị trong tương lai dựa vào giá trị trong quá khứ và hiện tại.
- Phân loại số ngẫu nhiên
  - Tất định: sinh bằng thuật toán sử dụng một mầm (seed) ngẫu nhiên; tuy tất định nhưng có tính chất của dãy ngẫu nhiên.
  - Bất định
    - Giả ngẫu nhiên: tổng hợp từ các đại lượng bất định
    - Ngẫu nhiên thực sự: sinh từ nguồn ngẫu nhiên vật lý



# Ứng dụng của số ngẫu nhiên

- Trong rất nhiều bài toán an toàn thông tin, người ta cần sử dụng số ngẫu nhiên, mà chính xác hơn là sử dụng các "chuỗi bit ngẫu nhiên"
  - Giá trị salt, challenge trong xác thực
  - Khóa mật mã, dù là khóa bí mật hay khóa công khai
  - Véc-tơ khởi tạo (IV) trong mã khối, mã dòng
  - Chuyển đổi thông điệp trong ký số và mã hóa khóa công khai
  - Sinh SessionID trong quản lý phiên
  - v.v..

# Weak Random Numbers

## ☐ Sự tương ứng với các CWE

- CWE-330: Use of Insufficiently Random Values
- CWE-331: Insufficient Entropy
- CWE-334: Small Space of Random Values
- CWE-335: PRNG Seed Error
- CWE-338: Use of Cryptographically Weak PRNG
- CWE-340: Predictability Problems
- CWE-341: Predictable from Observable State
- CWE-342: Predictable Exact Value from Previous Values
- CWE-343: Predictable Value Range from Previous Values

# Weak Random Numbers

## ☐ Sự tương ứng với các CWE

- CWE-330: Use of Insufficiently Random Values
- CWE-331: Insufficient Entropy
- CWE-334: Small Space of Random Values
- CWE-335: PRNG Seed Error
- CWE-338: Use of Cryptographically Weak PRNG
- CWE-340: Predictability Problems
- CWE-341: Predictable from Observable State
- CWE-342
- CWE-343

Nếu có thể đoán được session-id thì phiên làm việc  
có thể bị chiếm đoạt

# Random vs Secure Random

---

- Hãy đọc "**Random vs Secure Random numbers in Java**" (<https://www.geeksforgeeks.org/random-vs-secure-random-numbers-java/>)  
trước khi tiếp tục

# Sinh dãy ngẫu nhiên an toàn

Ngôn ngữ	An toàn (!?!?)	Không an toàn
C/C++	Sử dụng thư viện ngoài: openssl...	srand() + rand() std::random_device
Java	java.security.SecureRandom	java.util.Random
PHP	random_bytes() //PHP 7+	
.NET	System.Security.Cryptography. RandomNumberGenerator	System.Random
Linux	/dev/random (blocked) /dev/urandom (unblocked)	
Windows	CryptGenRandom, BCryptGenRandom	
Assembly	RDRAND (true random) RDSEED (pseudo-random)	

# **Using Cryptography Incorrectly**

# Using Cryptography Incorrectly

---

## ❑ Sự tương ứng với CWE

- CWE-326: Weak Encryption
- CWE-327: Use of a Broken or Risky Cryptographic Algorithm

# Các nhóm thuật toán mật mã

- Mật mã đối xứng
  - Mã khối (DES, AES, Blowfish,...)
  - Mã dòng (RC4, A5,...)
- Hàm băm (MD5, SHA1, SHA-256, SHA-512, Bcrypt...)
- Mật mã khóa công khai (DH, RSA, DSA, ECC,..)
- Sinh số ngẫu nhiên



# Các nhóm thuật toán mật mã

- Mật mã đối xứng
  - Mã khối (DES, AES, Blowfish,...)
  - Mã dòng (RC4, A5,...)
- Hàm băm (MD5, SHA1, SHA-256, SHA-512, Bcrypt...)
- Mật mã khóa công khai
- Sinh số ngẫu nhiên

Ứng dụng trong mã hóa dữ liệu

# Các nhóm thuật toán mật mã

- Mật mã đối xứng
  - Mã khối (DES, AES, Blowfish,...)
  - Mã dòng (RC4, A5,...)
- Hàm băm (MD5, SHA1, SHA-256, SHA-512, Bcrypt...)
- Mật mã khóa công khai (DH, RSA, DSA, ECC,...)
- Sinh số ngẫu nhiên
  - Ứng dụng rất phong phú
    - Kiểm tra toàn vẹn
    - Xác thực thông điệp, thực thể
    - Chữ ký số
    - ...

# Các nhóm thuật toán mật mã

- Mật mã đối xứng
  - Mã khối (DES, AES, Blowfish,...)
  - Mã dòng (RC4, A5,...)
- Hàm băm (MD5, SHA1, SHA-256, SHA-512, Bcrypt...)
- Mật mã khóa công khai (DH, RSA, DSA, ECC,..)
- Sinh số ngẫu nhiên

- Trao đổi khóa
- Mã hóa dữ liệu kích thước nhỏ
- Chữ ký số

# Các nhóm thuật toán mật mã

- Mật mã đối xứng
  - Mã khối (DES, AES, Blowfish,...)
  - Mã dòng (RC4, A5,...)
- Hàm băm (MD5, SHA1, SHA-256, SHA-512, Bcrypt...)
- Mật mã khóa công khai (DH, RSA, DSA, ECC,...)
- Sinh số ngẫu nhiên



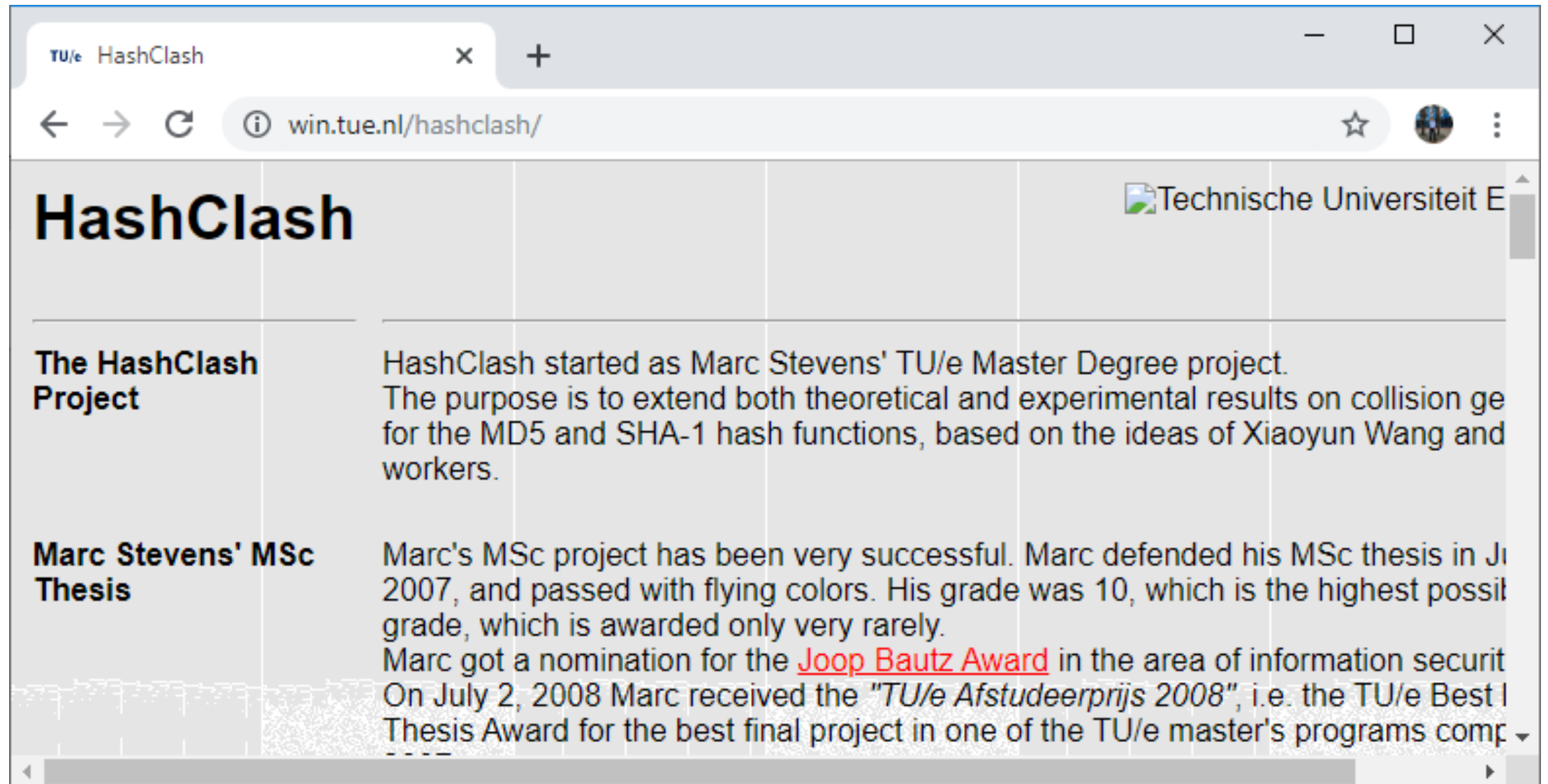
!?!?!?

**Có rất nhiều vấn đề an toàn trong  
lập trình sử dụng mật mã**

# Using Cryptography Incorrectly

- Sử dụng thuật toán không còn an toàn
- Sử dụng giao thức, lược đồ không an toàn
- Sinh tham số không an toàn
- Quản lý khóa mật mã không an toàn
- Tự thiết kế giao thức, thuật toán

# MD5 and SHA-1 Collision



The screenshot shows a web browser window with the title 'TU/e HashClash'. The address bar displays 'win.tue.nl/hashclash/'. The page content includes a header with the 'HashClash' title and the 'Technische Universiteit Eindhoven' logo. Below the header, there are two main sections: 'The HashClash Project' and 'Marc Stevens' MSc Thesis'. The 'The HashClash Project' section describes the project's origin as Marc Stevens' TU/e Master Degree project, aimed at extending theoretical and experimental results on collision generation for MD5 and SHA-1 hash functions, based on the ideas of Xiaoyun Wang and his workers. The 'Marc Stevens' MSc Thesis' section details the success of his MSc project, noting that he defended his thesis in June 2007 with a grade of 10, the highest possible. It also mentions his nomination for the 'Joop Bautz Award' in information security and his receipt of the 'TU/e Afstudeerprijs 2008' (TU/e Best Thesis Award) on July 2, 2008, for the best final project in one of the TU/e master's programs.

HashClash	
<b>The HashClash Project</b>	HashClash started as Marc Stevens' TU/e Master Degree project. The purpose is to extend both theoretical and experimental results on collision generation for the MD5 and SHA-1 hash functions, based on the ideas of Xiaoyun Wang and his workers.
<b>Marc Stevens' MSc Thesis</b>	Marc's MSc project has been very successful. Marc defended his MSc thesis in June 2007, and passed with flying colors. His grade was 10, which is the highest possible grade, which is awarded only very rarely. Marc got a nomination for the <a href="#">Joop Bautz Award</a> in the area of information security. On July 2, 2008 Marc received the "TU/e Afstudeerprijs 2008", i.e. the TU/e Best Thesis Award for the best final project in one of the TU/e master's programs completed.

# SHA-1 Collision

The screenshot shows a web browser window with the address bar displaying 'shattered.io'. The page is divided into two main sections: 'Attack proof' and 'File tester'.

**Attack proof**

Here are two PDF files that display different content, yet have the same SHA-1 digest.

Below this text, there are two side-by-side PDF thumbnails. Both thumbnails have a blue header with the text 'SHattered' and 'The first concrete collision attack against SHA-1'. The left thumbnail has a red footer with 'CWI' and 'Google' logos, and the right thumbnail has a red footer with 'CWI' and 'Google' logos. Both thumbnails list the names: Marc Stevens, Ange Albertini, and Yarik Markov.

Below the thumbnails, there is a terminal window showing the command 'sha1sum \*.pdf' and the output:

```
38762cf7f55934b34d179ae6a4c80cadccbb7f0a 1.pdf
38762cf7f55934b34d179ae6a4c80cadccbb7f0a 2.pdf
```

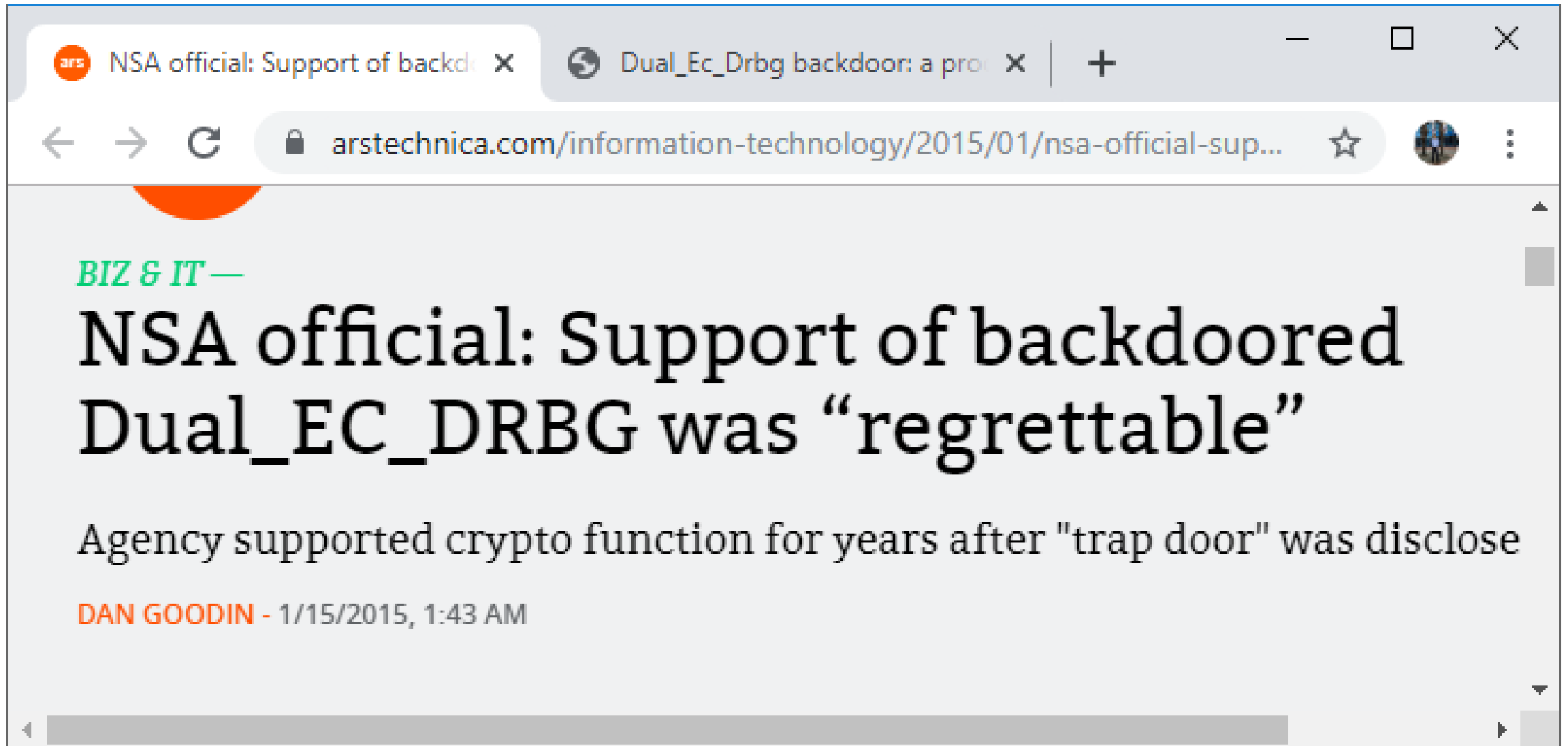
**File tester**

Upload any file to test if they are part of a collision attack. Rest assured that we do not store uploaded files.

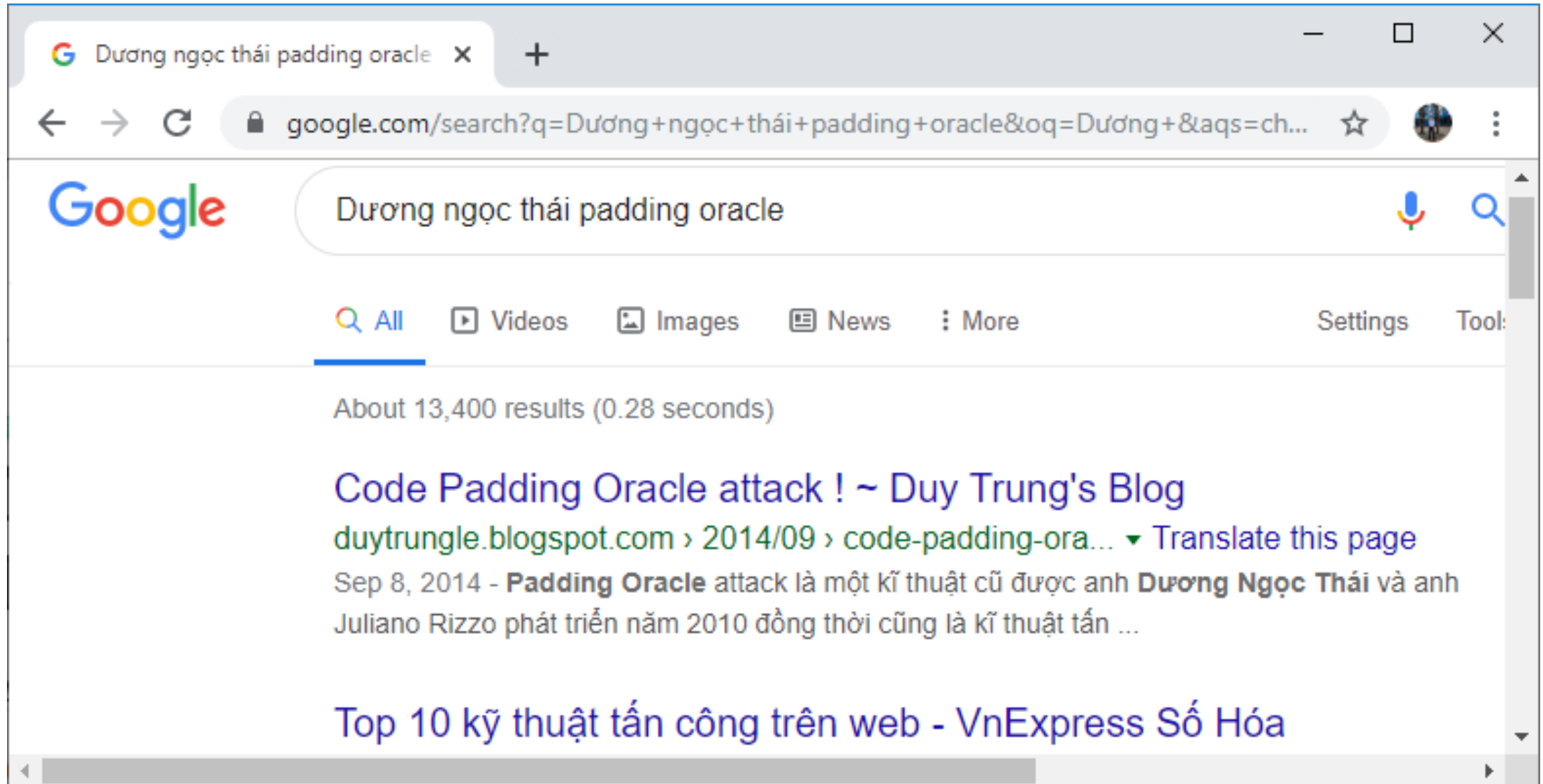
Below this text, there is a dashed box containing an upload icon (a circle with an upward arrow) and the text 'Drag some files here'.



# Dual\_EC\_DRBG



# Cipher Block Chaining



- OpenSSL
- Microsoft Windows CAPI (Cryptography API)
- Microsoft Windows CNG (CAPI Next Generation)
- JCA (Java Cryptography Architecture)
- ...

1

Lỗi hỏng do lập trình

2

Lỗi hỏng sử dụng mật mã

3

Lỗi hỏng mạng

4

Lỗi hỏng web

## **Failing to Protect Network Traffic**

**chỉ việc ứng dụng không có giải pháp thích hợp để  
bảo vệ thông tin truyền qua mạng**

# Hiểm họa đối với thông tin truyền qua mạng

---

- Khám phá → lộ bí mật
- Sửa đổi → mất toàn vẹn
- Phá hủy → tính khả dụng
- Tráo đổi, Mạo danh → tính xác thực

# Failing to Protect Network Traffic

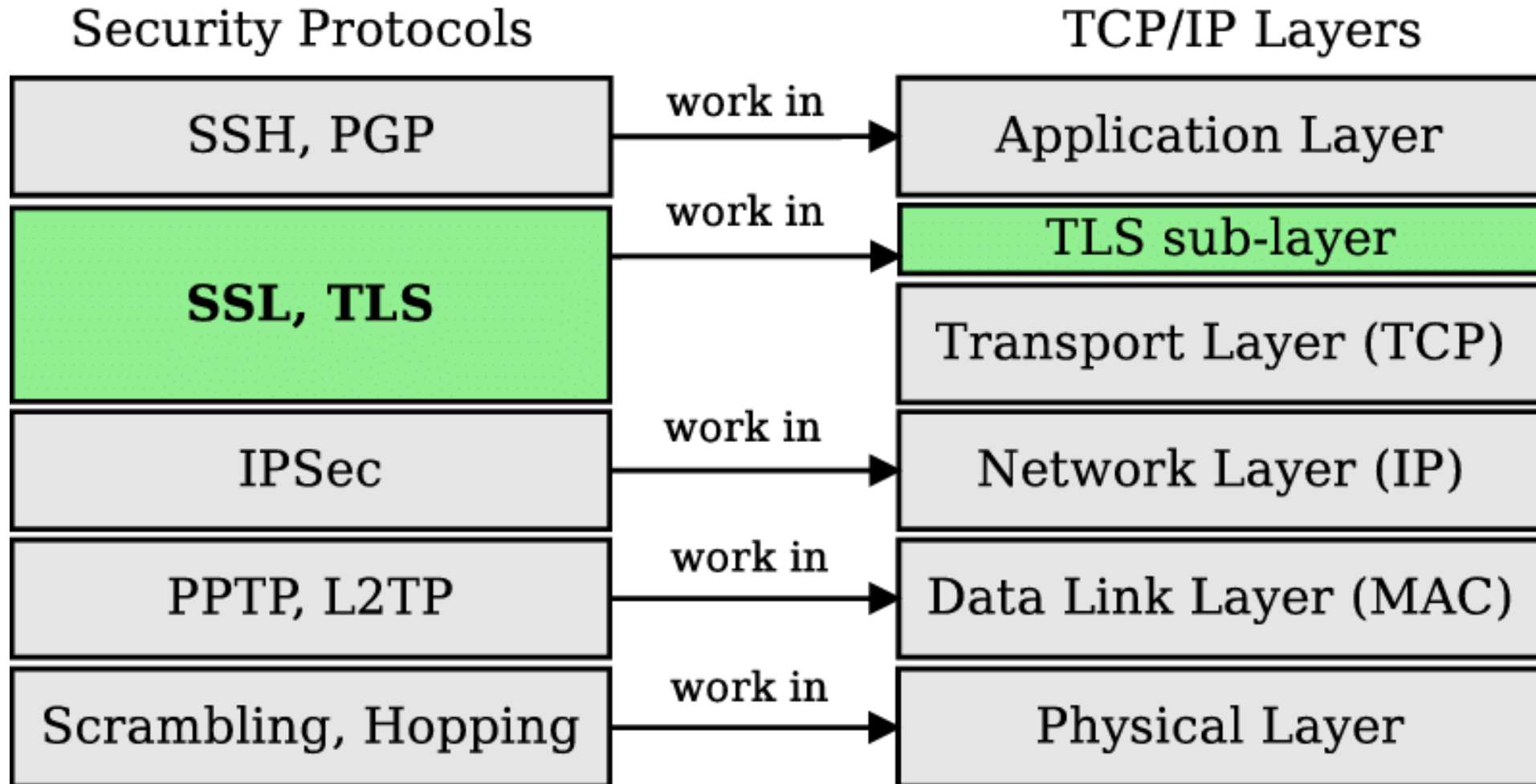
---

## ❑ Sự tương ứng với CWE

### ■ CWE-319: Cleartext Transmission of Sensitive Information

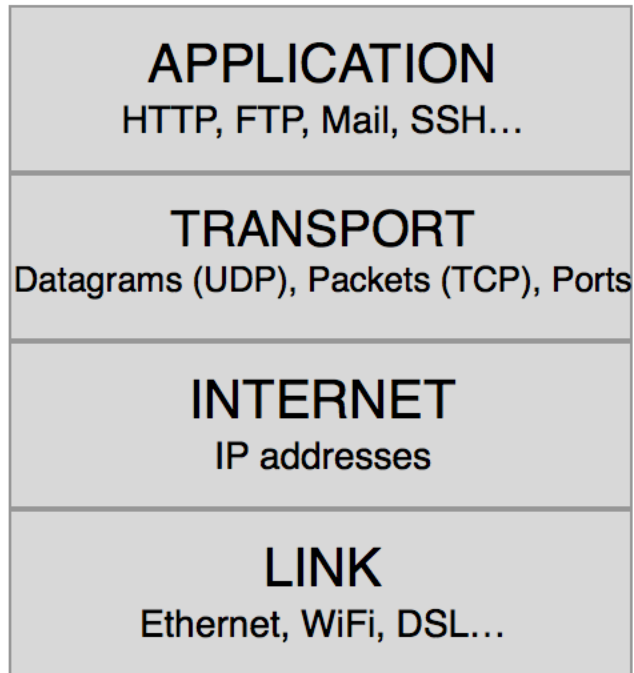
But “protection of data” is not limited to maintaining secrecy; you must also worry about tamper resistance and more

# Giao thức an toàn trên TCP/IP stack

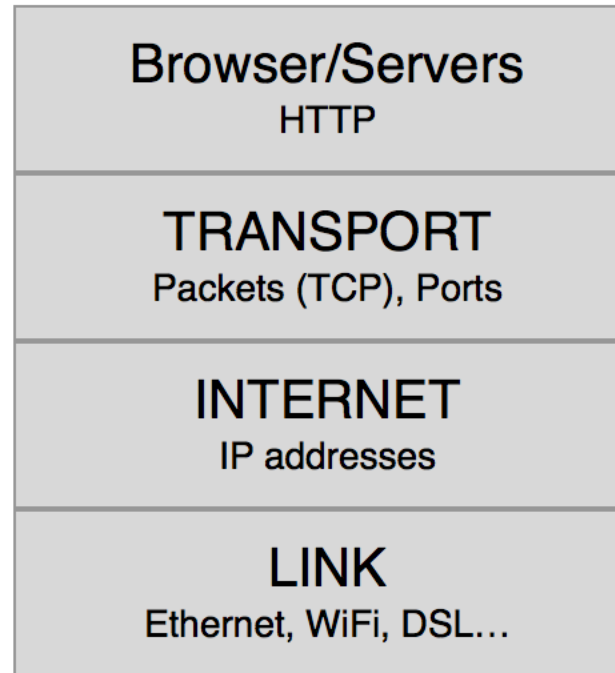




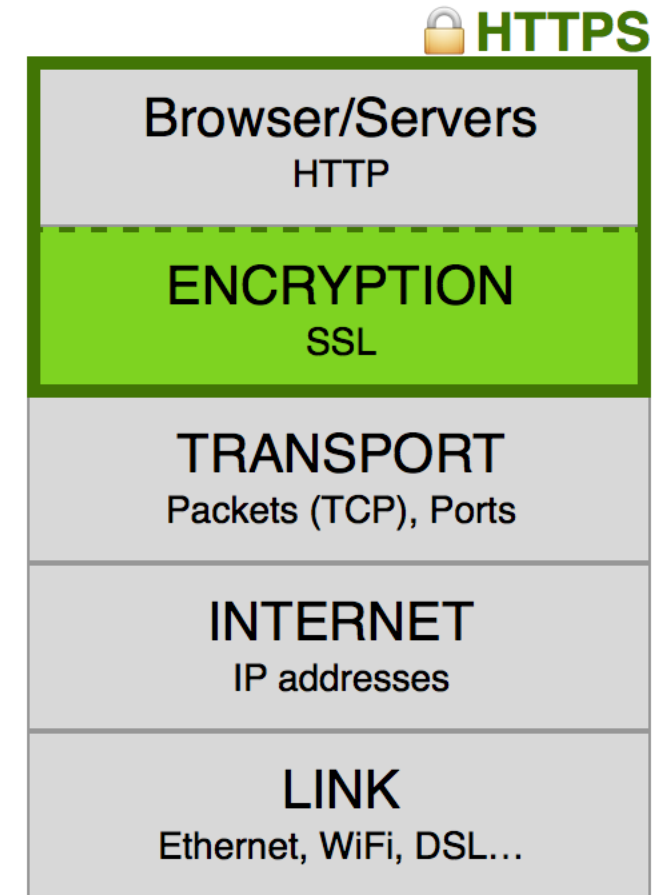
# Thực tiễn bảo vệ tầng ứng dụng



***Internet Protocol Stack***



***With HTTP***



***With HTTPS***

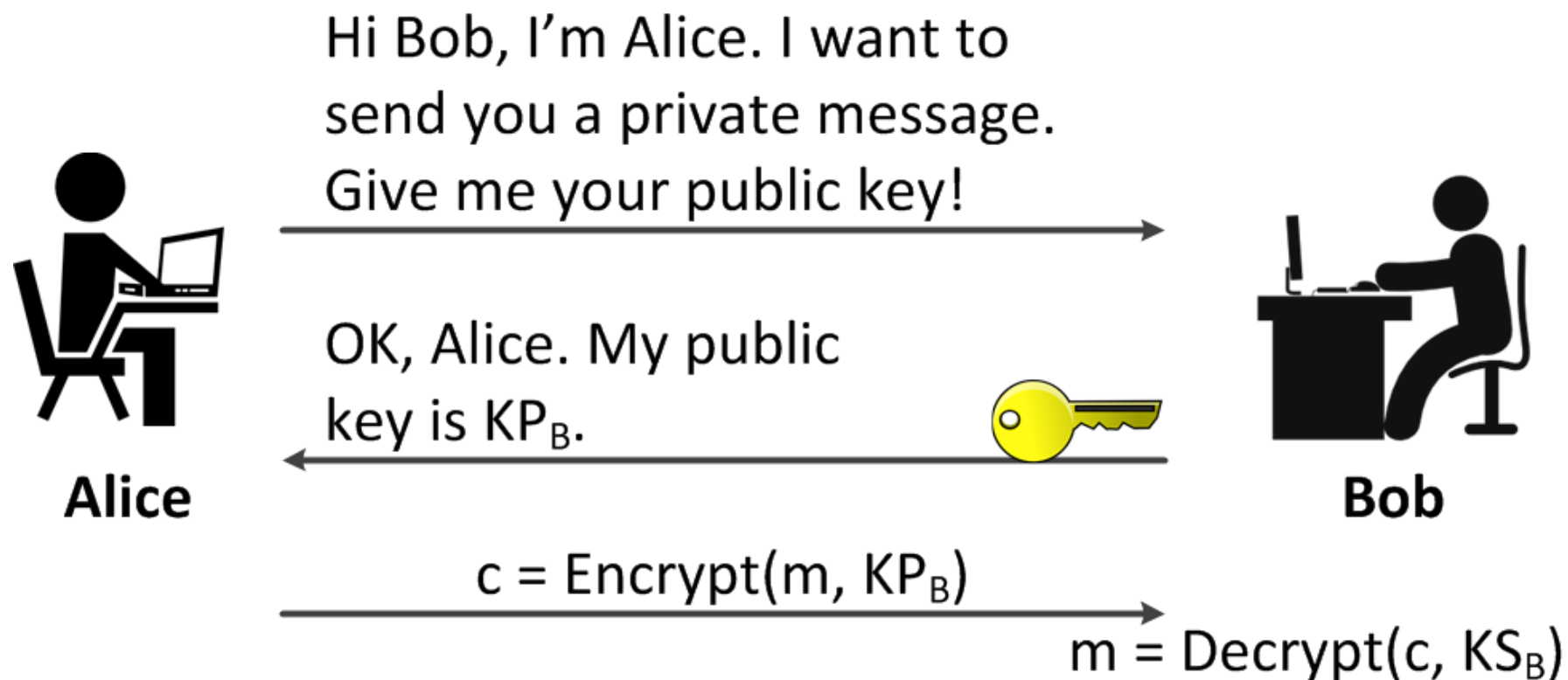
# Khuyến cáo an toàn

- TLS (Transport Layer Security)
  - bí mật, toàn vẹn và xác thực
  - là lớp đệm giữa tầng Application và Transport
  - không trong suốt đối với tầng ứng dụng (!)
- Nhiều giao thức không an toàn đã được nâng cấp, sử dụng TLS để đảm bảo an toàn: POP3, SMTP, HTTP, FTP...
- Các thư viện mật mã đã hỗ trợ giao thức này.
- Hãy sử dụng TLS nếu triển khai giao tiếp mạng!

## **Improper Use of PKI, Especially SSL**

**chỉ những lỗi hổng do thiếu sót trong kiểm tra tính  
hợp lệ của chữ ký số**

# Sự tuyệt vời của mật mã khóa công khai



# Thực ra thì... không tuyệt vời lắm (!)

**Alice**



Hi Bob, I'm Alice. Send me your public key!

OK, Alice, my public key is  $KP_M$

$c = \text{Enc}(m, KP_M)$

**Malice**



Hi Bob, I'm Alice. Send me your public key!

OK, Alice, my public key is  $KP_B$

$c' = \text{Enc}(m, KP_B)$

**Bob**



# Phải có PKI thì mới thực sự tuyệt vời

## CHỨNG THƯ KHÓA CÔNG KHAI

Tôi là: **Trent**

## CHỨNG NHẬN

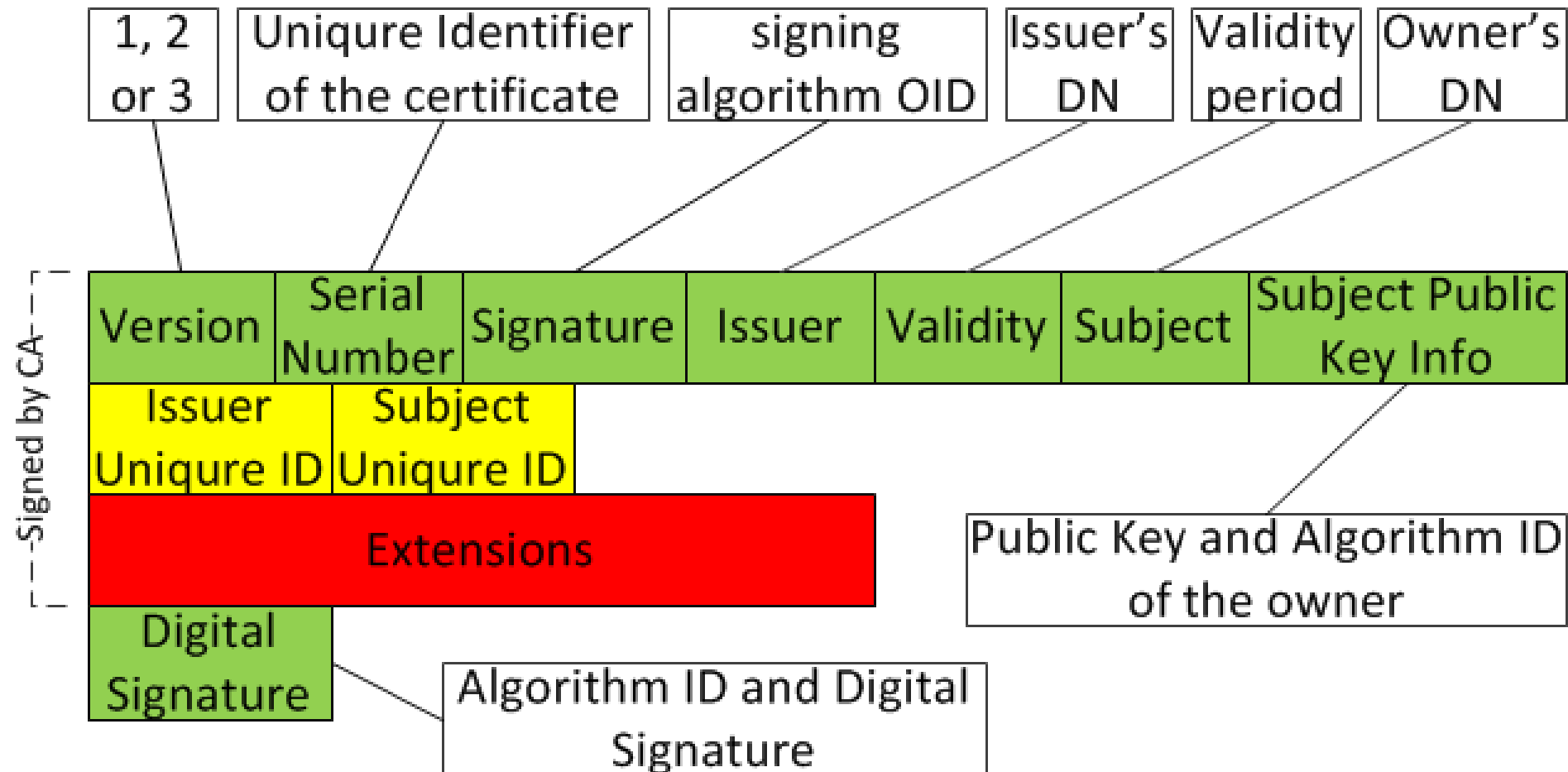
Ông/bà: **Bob**

Có khóa công khai là: 010101011

(Trent đã kí)

1101010111

# Nhưng X.509 v3 không hề đơn giản



## Việc kiểm tra tính hợp lệ vì thế cũng không đơn giản

- Kiểm tra thời hạn có hiệu lực
- Kiểm tra tính hợp lệ của chữ kí của CA
  - Phải có (chuỗi) Certificate **tin cậy** của CA (!)
- Đối chiếu mục đích sử dụng chứng thư
- Kiểm tra trạng thái thu hồi chứng thư



# Improper Use of PKI, Especially SSL

## ❑ Sự tương ứng với CWE

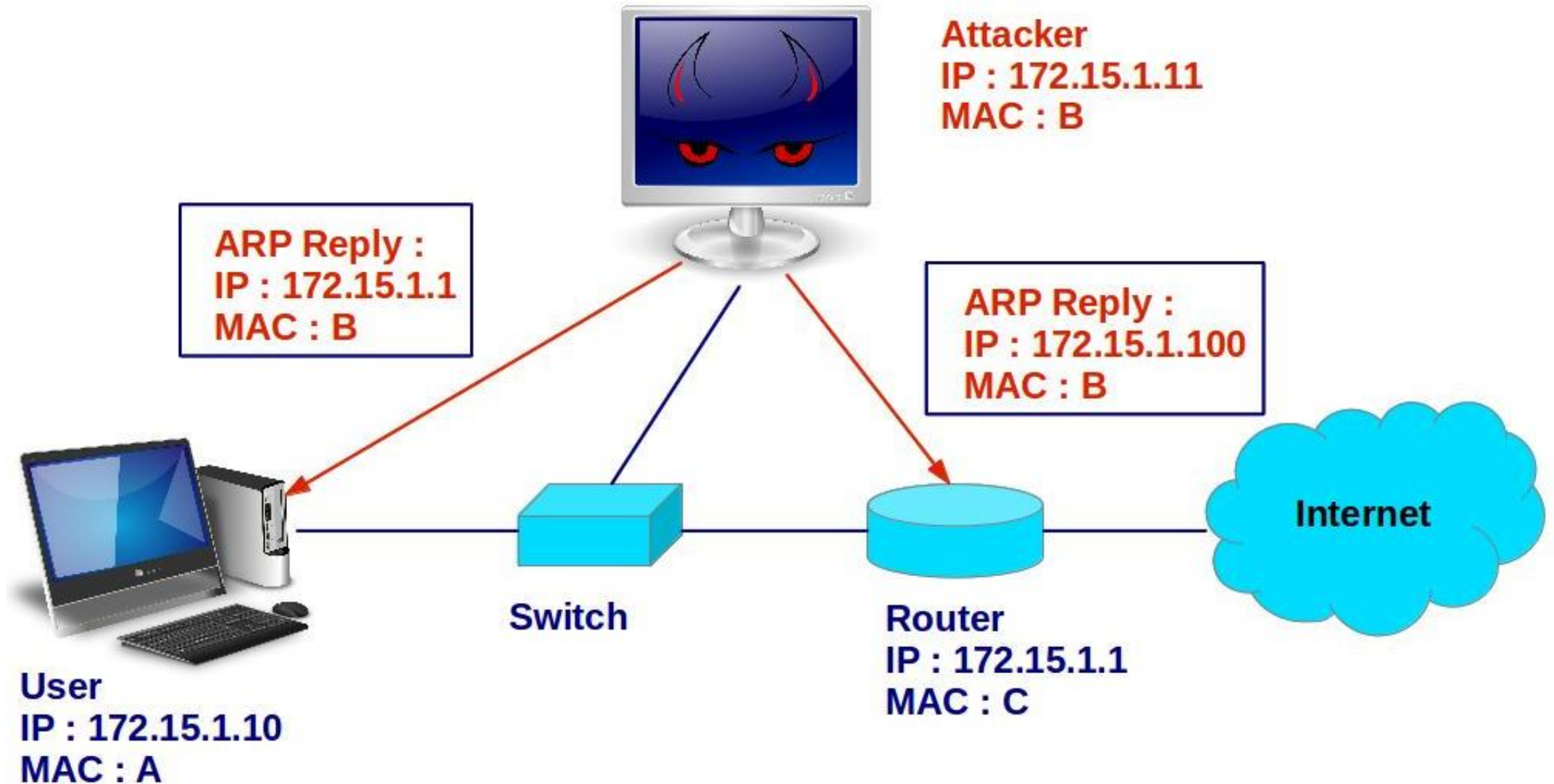
- CWE-296: Failure to Follow Chain of Trust in Certificate Validation
- CWE-297: Failure to Validate Host-Specific Certificate Data
- CWE-298: Failure to Validate Certificate Expiration
- CWE-299: Failure to Check for Certificate Revocation
- CWE-324: Use of a Key Past Its Expiration Date

## ❑ Giải pháp ngăn ngừa

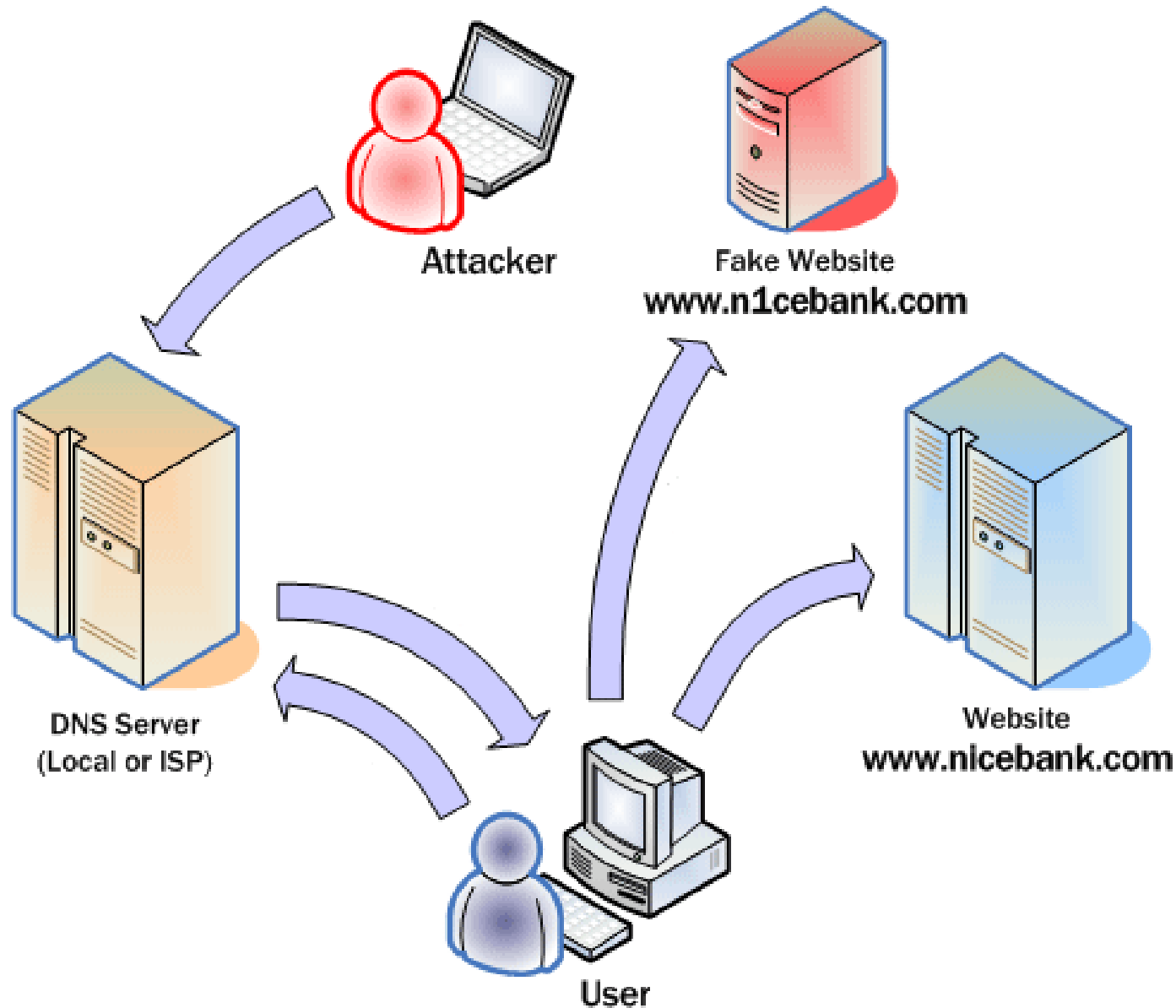
- Cố gắng tránh việc tự thực hiện. Hãy sử dụng hàm thư viện bậc cao khi có thể.
- Ví dụ: lớp **X509Certificate2** trong .NET Framework

**Trusting Network Name Resolution** chỉ  
những lỗ hổng do không kiểm tra tính tin  
cậy của máy chủ DNS

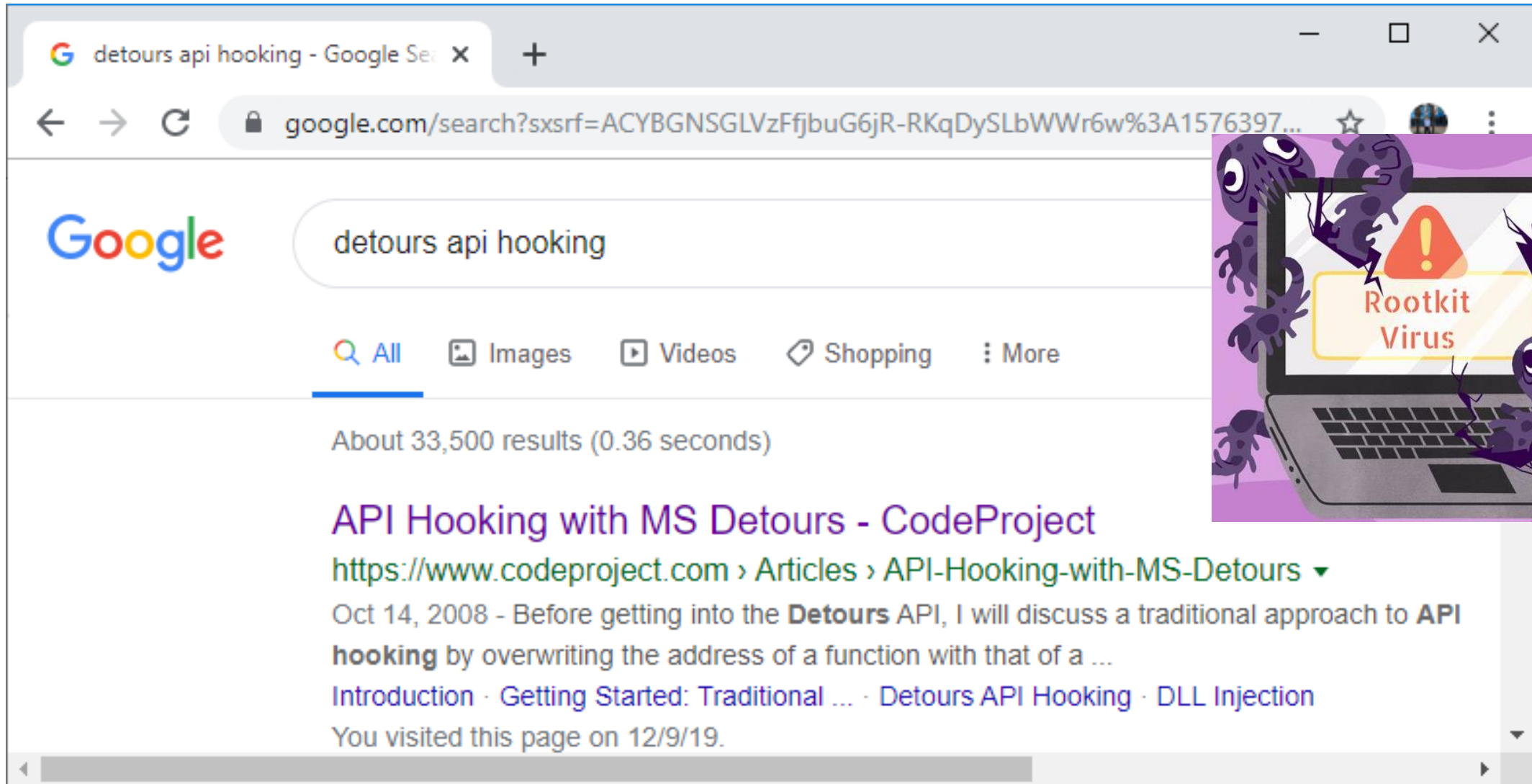
# ARP Poisoning can...



# ... lead to DNS Spoofing

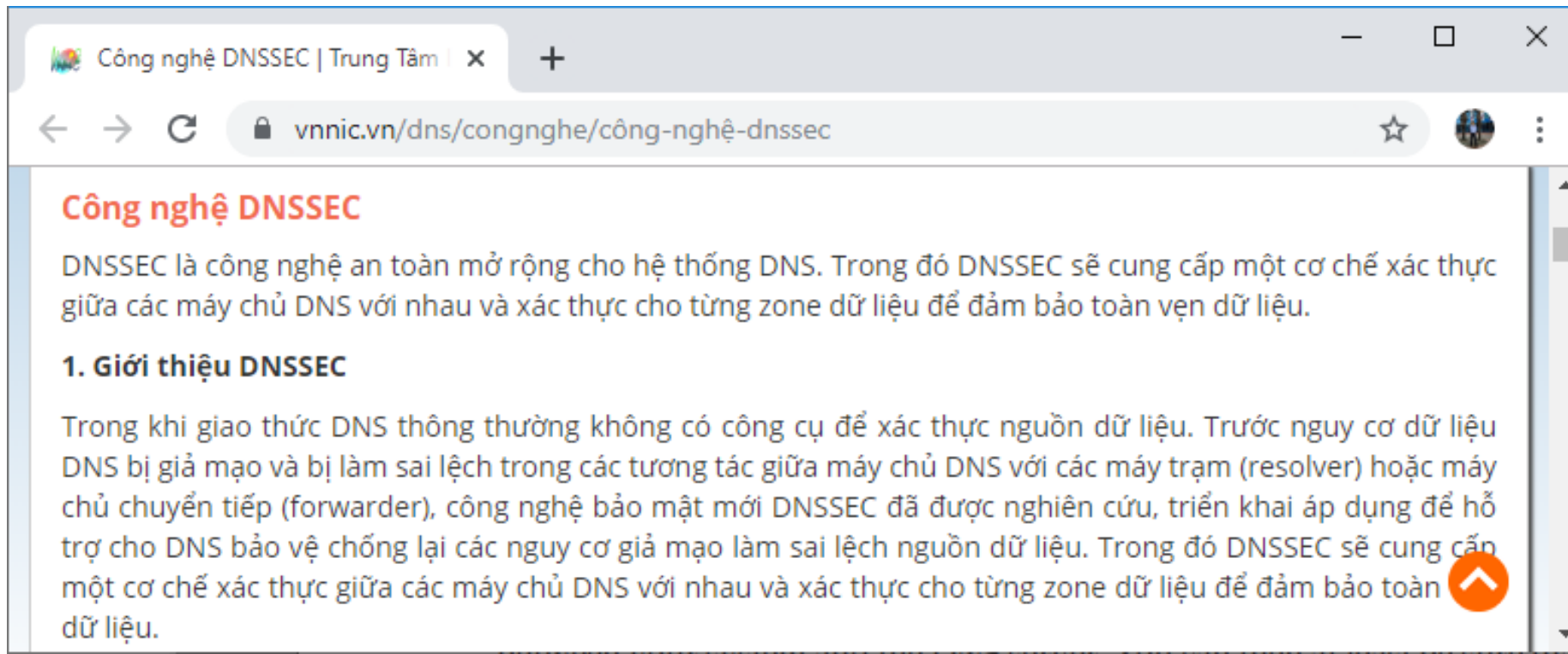


# Không chỉ có thể



# Trusting Network Name Resolution

- Giải pháp: DNSSEC (Domain Name System Security Extensions)
  - DNSSEC là công nghệ an toàn mở rộng cho hệ thống DNS. Trong đó DNSSEC sẽ cung cấp một cơ chế xác thực giữa các máy chủ DNS với nhau và xác thực cho từng zone dữ liệu để đảm bảo toàn vẹn dữ liệu.



1

Lỗi hỏng do lập trình

2

Lỗi hỏng sử dụng mật mã

3

Lỗi hỏng mạng

4

Lỗi hỏng web

