

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
KHOA AN TOÀN THÔNG TIN**



**BÁO CÁO THỰC HÀNH  
HỌC PHẦN: MẬT MÃ HỌC CƠ SỞ  
MÃ HỌC PHẦN: INT1344**

**BÀI THỰC HÀNH:  
Giao thức Diffie-Hellman**

Sinh viên thực hiện:

Nguyễn Văn Hùng - B22DCAT136

Giảng viên hướng dẫn: TS. Quản Trọng Thế

**HÀ NỘI 5-2025**

# MỤC LỤC

MỤC LỤC .....	1
DANH MỤC CÁC HÌNH VẼ .....	2
DANH MỤC CÁC BẢNG BIỂU .....	2
DANH MỤC CÁC TỪ VIẾT TẮT .....	3
CHƯƠNG 1. Giới thiệu chung về bài thực hành .....	4
1.1 Mục đích .....	4
1.2 Tìm hiểu lý thuyết .....	4
1.2.1 Giới thiệu .....	4
1.2.2 Cách hoạt động .....	4
1.2.3 Nguyên lý toán học .....	4
1.2.4 Ưu điểm và hạn chế .....	5
1.2.5 Ứng dụng thực tế .....	5
1.3 Kết chương .....	5
CHƯƠNG 2. Nội dung thực hành .....	6
2.1 Chuẩn bị môi trường .....	6
2.2 Các bước thực hiện .....	6
CHƯƠNG 3. Kết quả .....	8
TÀI LIỆU THAM KHẢO .....	9

## **DANH MỤC CÁC HÌNH VẼ**

Hình 1 . Khởi động bài lab .....	6
Hình 2 . Thực hành .....	7
Hình 3 . Kết quả bài lab .....	8

## **DANH MỤC CÁC BẢNG BIỂU**

Bảng 1. Ưu điểm và hạn chế của giao thức Diffie-Hellman .....	5
---	---

## DANH MỤC CÁC TỪ VIẾT TẮT

<b>Từ viết tắt</b>	<b>Thuật ngữ tiếng Anh/Giải thích</b>	<b>Thuật ngữ tiếng Việt/Giải thích</b>
TLS	Transport Layer Security	Bảo mật tầng truyền tải
SSL	Secure Sockets Layer	Lớp socket bảo mật
IPsec	Internet Protocol Security	Bảo mật giao thức Internet
IKE	Internet Key Exchange	Trao đổi khóa Internet
SSH	Secure Shell	Giao thức điều khiển từ xa an toàn
PGP	Pretty Good Privacy	Bảo mật khá tốt (PGP)
MAC	Message Authentication Code	Mã xác thực thông điệp

# CHƯƠNG 1. GIỚI THIỆU CHUNG VỀ BÀI THỰC HÀNH

## 1.1 Mục đích

- Giúp sinh viên hiểu cơ chế hoạt động của giao thức Diffie-Hellman trong việc trao đổi khóa.
- Cung cấp kiến thức và kỹ năng về bảo mật thông tin thông qua việc thực hành triển khai giao thức Diffie-Hellman.
- Sinh viên có thể áp dụng giao thức Diffie-Hellman trong các ứng dụng bảo mật thực tế.

## 1.2 Tìm hiểu lý thuyết

### 1.2.1 Giới thiệu

Giao thức Diffie–Hellman do Whitfield Diffie và Martin Hellman đề xuất năm 1976, thường được gọi là Alice và Bob, cho phép hai bên xác lập khóa chung một cách an toàn qua kênh công khai mà không cần chia sẻ trước khóa bí mật — điểm khởi đầu cho lĩnh vực mã hóa khóa công khai hiện đại. Phương pháp này giải quyết vấn đề truyền khóa an toàn qua đường truyền không tin cậy, giúp bảo vệ thông tin nhạy cảm trước kẻ nghe trộm

### 1.2.2 Cách hoạt động

Trao đổi khóa Diffie–Hellman bao gồm các bước sau:

- Alice và Bob thỏa thuận hai số nguyên tố lớn  $p$  (modulus) và  $g$  (nguyên gốc nguyên thủy modulo  $p$ ), hai giá trị này được công khai.
- Alice chọn một số ngẫu nhiên bí mật  $a$ , rồi tính  $A = g^a \bmod p$ , sau đó gửi  $A$  cho Bob.
- Bob chọn một số ngẫu nhiên bí mật  $b$ , rồi tính  $B = g^b \bmod p$  sau đó gửi  $B$  cho Alice.
- Alice tính khóa chung bí mật:  $s = B^a \bmod p$
- Bob tính khóa chung bí mật:  $s = A^b \bmod p$

Kết thúc quá trình, cả Alice và Bob đều có chung khóa bí mật  $s$  mà không truyền trực tiếp qua kênh không an toàn. Kẻ nghe trộm, dù biết  $p$ ,  $g$ ,  $A$  và  $B$ , cũng không thể tính được  $s$  một cách hiệu quả do bài toán logarit rời rạc rất khó giải.

### 1.2.3 Nguyên lý toán học

Dựa trên lũy thừa theo module  $A = g^a \bmod p$ ,  $B = g^b \bmod p$

Tính khóa chung  $s$  qua  $s = B^a \bmod p = A^b \bmod p$

Bảo mật nhờ bài toán logarit rời rạc: khó tìm  $a$  hay  $b$  từ  $A$ ,  $B$ ,  $p$ ,  $g$

### 1.2.4 Ưu điểm và hạn chế

Bảng 1. Ưu điểm và hạn chế của giao thức Diffie-Hellman

Ưu điểm	Nhược điểm
Bảo mật tiến (forward secrecy): Mỗi phiên giao tiếp tạo ra khóa chung mới, nên nếu một khóa bị lộ, các phiên trước và sau đó vẫn an toàn.	Dễ bị tấn công trung gian (man-in-the-middle): Giao thức không xác thực danh tính, nên kẻ tấn công có thể giả mạo để chặn hoặc thay đổi thông tin. Để khắc phục, thường kết hợp với chữ ký số hoặc cơ chế xác thực khác.
Khả năng mở rộng: Số lượng phép tính không tăng nhiều khi số bên tham gia tăng, mỗi bên chỉ phải thực hiện vài phép lũy thừa.	Chi phí tính toán: Phép lũy thừa theo mô đun với số nguyên tố lớn có thể tốn thời gian, nhưng có thể giảm bằng thuật toán lũy thừa nhanh hoặc sử dụng biến thể trên đường cong elliptic, cho khóa nhỏ hơn mà vẫn đảm bảo an toàn tương đương.
Không yêu cầu hai bên phải chia sẻ thông tin bí mật hay thiết lập độ tin cậy trước, phù hợp với các tình huống khó thiết lập kênh tin cậy ban đầu.	Không mã hóa dữ liệu hay đảm bảo tính toàn vẹn: Giao thức chỉ thiết lập khóa chung, không mã hóa hay bảo vệ tính toàn vẹn dữ liệu. Để bảo mật toàn diện, khóa chung phải kết hợp với thuật toán mã đối xứng và mã xác thực thông điệp (MAC) hoặc cơ chế mã hóa có xác thực.

### 1.2.5 Ứng dụng thực tế

- TLS/SSL, IPsec và nhiều giao thức bảo mật sử dụng Diffie–Hellman để thiết lập khóa phiên ban đầu.
- Internet Key Exchange (IKE) trong IPsec kết hợp Diffie–Hellman với xác thực bằng chứng chỉ để đạt an toàn cao hơn
- Ứng dụng trong mật mã đối xứng, như SSH, PGP, và các hệ thống mã hóa thương mại, tận dụng khóa chung để mã hóa dữ liệu truyền tải

## 1.3 Kết chương

Trong bài thực hành này, chúng ta đã tìm hiểu cơ sở toán học của giao thức Diffie–Hellman - dựa trên phép lũy thừa theo mô–đun và bài toán logarit rời rạc - cũng như quy trình trao đổi khóa an toàn qua kênh công khai. Đồng thời cũng có hiểu biết cơ sở về ưu điểm, hạn chế và một số ứng dụng thực tế của giao thức Diffie–Hellman.

## CHƯƠNG 2. NỘI DUNG THỰC HÀNH

### 2.1 Chuẩn bị môi trường

- Phần mềm ảo hóa VMWare Workstation
- Máy ảo Labtainer

### 2.2 Các bước thực hiện

Tải bài Lab:

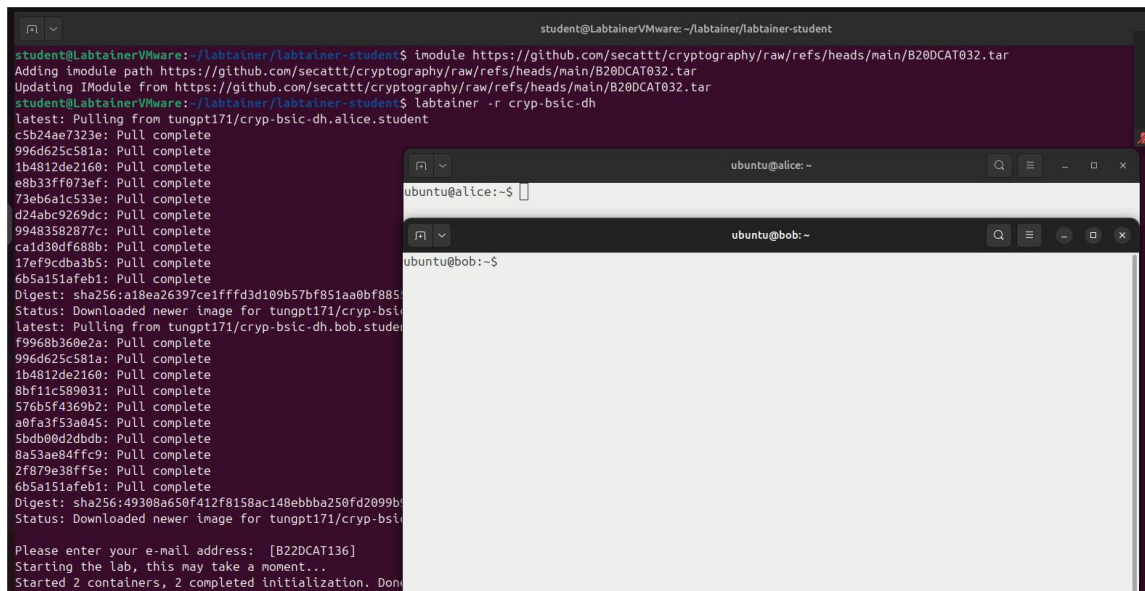
*imodule*

*<https://github.com/secattt/cryptography/raw/refs/heads/main/B20DCAT032.tar>*

Khởi động bài lab: *labtainer -r cryp-bsic-dh*

Nhập email là mã sinh viên: *B22DCAT136*

Hai terminal ảo sẽ được mở ra: máy server (Alice) và máy client (Bob).



Hình 1. Khởi động bài lab

Tiến hành trao đổi khóa Diffie-Hellman:

- Trên máy Server (Alice): *python3 server.py*
- Trên máy Client (Bob): *python3 client.py*
- Tại máy Client, nhập giá trị  $p$  (là một số nguyên tố lớn) dùng làm tham số công khai: 23 và nhập số nguyên  $g$ : 5
- Nhập private key:  $a=4$ ,  $b=6$
- Tính khóa công khai:
  - $A = g^a \bmod p = 5^4 \bmod 23 = 8$
  - $B = g^b \bmod p = 5^6 \bmod 23 = 4$

- Gửi và nhận khóa công khai từ server
- Tính khóa chung và xác minh:

$$\circ s = B^a \bmod p = A^b \bmod p = 8^4 \bmod 23 = 2$$

```

ubuntu@alice:~$ python3 server.py
Server is listening on 0.0.0.0:12345
Connection success with ('172.10.0.6', 54490)
Receive public p,g from Bob: p = 23, g = 5
Enter the private key a (for g = 5, p = 23): 4
4
Calculate the public key A: 4
4
Correct! Proceeding with public key exchange...
Send public key A to Bob...
Receive public key B from Bob: 8
Now, calculate the shared secret:
Calculate the shared secret key based on B, a, p: 2
2
Correct! Shared secret key validated.
Shared key: 2
ubuntu@alice:~$

ubuntu@bob:~$ python3 client.py
Enter value for p (prime number): 23
23
Enter value for g (generator): 5
5
Enter the private key b (for g = 5, p = 23): 6
6
Calculate the public key B: 8
8
Correct! Proceeding with public key exchange...
Send Public_key to Alice...
Received public key A from Alice: 4
Now, calculate the shared secret key based on the formula
Calculate the shared secret key based A, b, p: 2
2
Correct! Shared secret key validated.
Shared key: 2
ubuntu@bob:~$

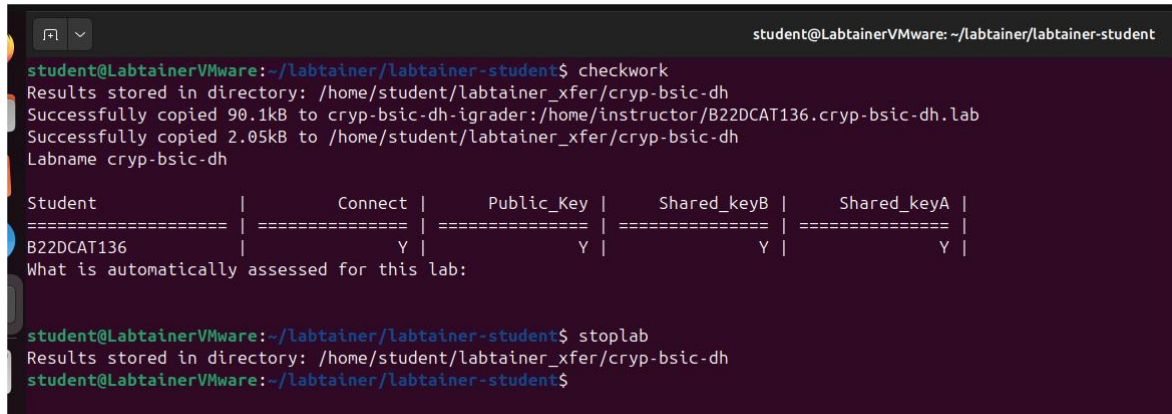
```

*Hình 2. Thực hành*



## CHƯƠNG 3. KẾT QUẢ

- Kết thúc bài lab: *stoplab*



```
student@LabtainerVMware:~/labtainer/labtainer-student$ checkwork
Results stored in directory: /home/student/labtainer_xfer/cryp-bsic-dh
Successfully copied 90.1kB to cryp-bsic-dh-igrader:/home/instructor/B22DCAT136.cryp-bsic-dh.lab
Successfully copied 2.05kB to /home/student/labtainer_xfer/cryp-bsic-dh
Labname cryp-bsic-dh

Student      | Connect | Public_Key | Shared_keyB | Shared_keyA |
=====|=====|=====|=====|=====|
B22DCAT136   | Y       | Y         | Y         | Y         |
What is automatically assessed for this lab:

student@LabtainerVMware:~/labtainer/labtainer-student$ stoplab
Results stored in directory: /home/student/labtainer_xfer/cryp-bsic-dh
student@LabtainerVMware:~/labtainer/labtainer-student$
```

Hình 3. Kết quả bài lab

- Khởi động lại lab nếu cần: *labtainer -r cryp-bsic-dh*
- Kết quả bài lab được lưu: */home/student/labtainer\_xfer/cryp-bsic-dh*

## TÀI LIỆU THAM KHẢO

- [1] <https://www.1kosmos.com/security-glossary/diffie-hellman-key-exchange-algorithm/>