TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

**KHOA CÔNG NGHỆ THÔNG TIN**

------🙙🕮🙛-------



**BÁO CÁO THỰC NGHIỆM**

Học phần: An ninh mạng

**Chủ đề**: **Chuẩn DES và ứng dụng trong mã hóa dữ liệu (Sử dụng ngôn ngữ C#,Java)**

Giáo viên hướng dẫn : TS. Phạm Văn Hiệp

Nhóm sinh viên thực hiện :

1. Trần Trọng Hiếu Mã SV: 2022602004

2. Lê Đức Chương Mã SV: 2022602955

3.Nguyễn Viết Minh Hoàng Mã SV: 2022600282

4.Trần Xuân Hoàn Mã SV: 2022601562

5.Lê Văn Giang Mã SV: 2021606816

Mã Lớp học phần:

Nhóm: 4

Hà Nội - Năm 2024

**MỤC LỤC**

Danh mục các từ viết tắt…

Danh mục hình ảnh…

Danh mục bảng biểu…

…Các chương, các đề mục trong báo cáo…

# **Lời nói đầu**

Trong thời đại công nghệ số, dữ liệu không chỉ đơn thuần là thông tin, mà còn là tài sản vô giá của cá nhân, tổ chức và quốc gia. Chính vì vậy, việc bảo vệ dữ liệu trước các nguy cơ như truy cập trái phép, đánh cắp, và giả mạo ngày càng trở nên quan trọng. Một trong những giải pháp hữu hiệu và phổ biến nhất để đảm bảo tính bảo mật cho dữ liệu chính là áp dụng các thuật toán mã hóa.

Thuật toán mã hóa DES (Data Encryption Standard), mặc dù đã xuất hiện từ lâu, vẫn giữ vai trò quan trọng trong lịch sử phát triển các công nghệ mã hóa hiện đại. DES không chỉ là nền tảng cho nhiều thuật toán mã hóa phức tạp hơn mà còn là công cụ học thuật quan trọng để sinh viên nghiên cứu về nguyên lý hoạt động của các hệ thống bảo mật thông tin.

Nhận thức được tầm quan trọng của chủ đề này, nhóm chúng em đã lựa chọn thực hiện bài tập lớn môn An ninh mạng với chủ đề: “Chuẩn DES và ứng dụng trong mã hóa dữ liệu”. Mục tiêu của bài tập lớn không chỉ dừng lại ở việc tìm hiểu lý thuyết mà còn tập trung vào triển khai thực tiễn thuật toán DES bằng hai ngôn ngữ lập trình phổ biến là C# và Java, nhằm xây dựng nền tảng kiến thức vững chắc cũng như rèn luyện kỹ năng lập trình.

Bài báo cáo là kết quả của quá trình nghiên cứu, phân tích tài liệu, và triển khai thực tiễn trong suốt thời gian thực hiện bài tập lớn. Trong quá trình thực hiện, chúng em đã cố gắng học hỏi, nỗ lực hoàn thành bài tập với sự nghiêm túc cao nhất. Tuy nhiên, do hạn chế về thời gian và kinh nghiệm, chắc chắn bài làm sẽ không tránh khỏi những thiếu sót.

Chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của giáo viên bộ môn An ninh mạng, và sự hỗ trợ của các tài liệu tham khảo trong quá trình nghiên cứu.

Hy vọng rằng bài báo cáo này không chỉ đáp ứng yêu cầu của môn học mà còn góp phần nâng cao kiến thức của chúng em về lĩnh vực an ninh mạng, từ đó ứng dụng hiệu quả vào công việc thực tế trong tương lai.

# **Chương 1: Tổng quan**

## **1.1 Tổng quan về An ninh mạng**

### **1.1.1 Khái niệm về an ninh mạng**

An ninh mạng, hay còn gọi là bảo mật mạng (Cybersecurity), là tập hợp các phương pháp, công nghệ và quy trình được thiết kế nhằm bảo vệ hệ thống mạng, thiết bị, và dữ liệu khỏi các cuộc tấn công, truy cập trái phép, hoặc phá hoại. Trong thời đại công nghệ 4.0, khi thông tin được số hóa và lưu trữ trực tuyến, an ninh mạng đóng vai trò quan trọng trong việc đảm bảo sự an toàn và ổn định của các tổ chức, doanh nghiệp, cũng như cá nhân.

An ninh mạng không chỉ bảo vệ dữ liệu khỏi các mối đe dọa mà còn đảm bảo các nguyên tắc cơ bản:

* Tính bí mật (Confidentiality): Chỉ những đối tượng được cấp quyền mới có thể truy cập thông tin.
* Tính toàn vẹn (Integrity): Dữ liệu không bị sửa đổi hoặc phá hoại bởi những tác nhân trái phép.
* Tính sẵn sàng (Availability): Hệ thống và dữ liệu luôn sẵn sàng để truy cập khi cần thiết.

### **1.1.2. Vai trò của an ninh mạng**

An ninh mạng có vai trò thiết yếu trong tất cả các lĩnh vực, đặc biệt là trong:

* Chính phủ và quốc phòng: Bảo vệ các thông tin mật, ngăn chặn các cuộc tấn công mạng có mục tiêu chính trị.
* Doanh nghiệp: Đảm bảo thông tin khách hàng, dữ liệu nội bộ không bị rò rỉ, giữ vững uy tín và tính cạnh tranh.
* Người dùng cá nhân: Bảo vệ thông tin cá nhân, tài khoản ngân hàng, và tránh các hành vi lừa đảo trực tuyến.

Trong bối cảnh các cuộc tấn công mạng ngày càng gia tăng về số lượng và mức độ tinh vi, an ninh mạng không chỉ là vấn đề kỹ thuật mà còn là yếu tố sống còn để phát triển kinh tế và bảo vệ quyền lợi người dùng.

### **1.1.3. Các mối đe dọa an ninh mạng phổ biến**

Hiện nay, có nhiều loại mối đe dọa an ninh mạng, từ các cuộc tấn công nhỏ lẻ đến các chiến dịch tấn công quy mô lớn:

* Malware (Phần mềm độc hại): Bao gồm virus, ransomware, spyware, và trojan, thường được sử dụng để chiếm quyền kiểm soát hoặc làm hỏng hệ thống.
* Phishing (Tấn công lừa đảo): Lợi dụng sự tin tưởng của người dùng để thu thập thông tin nhạy cảm như tài khoản ngân hàng hoặc mật khẩu.
* DDoS (Tấn công từ chối dịch vụ): Làm gián đoạn hoạt động của hệ thống bằng cách gửi lượng lớn yêu cầu khiến hệ thống bị quá tải.
* APT (Mối đe dọa kéo dài và tinh vi): Các cuộc tấn công nhắm đến mục tiêu cụ thể, thường nhằm đánh cắp dữ liệu quan trọng hoặc phá hoại hệ thống.
* Tấn công vào chuỗi cung ứng: Nhắm vào các nhà cung cấp hoặc đối tác của mục tiêu để xâm nhập hệ thống.

### **1.1.4. Các phương pháp bảo vệ an ninh mạng**

Để đối phó với các mối đe dọa trên, nhiều giải pháp và công nghệ đã được triển khai:

* Tường lửa (Firewall): Ngăn chặn lưu lượng truy cập trái phép vào hệ thống mạng.
* Mã hóa dữ liệu: Sử dụng các thuật toán như DES, AES, RSA để đảm bảo thông tin được bảo vệ trong quá trình truyền tải.
* Hệ thống phát hiện và ngăn chặn xâm nhập (IDS/IPS): Giám sát hoạt động mạng để phát hiện và ngăn chặn các mối đe dọa.
* Xác thực hai yếu tố (2FA): Tăng cường bảo mật bằng cách yêu cầu người dùng xác minh danh tính qua nhiều bước.
* Cập nhật hệ thống thường xuyên: Vá lỗi và cập nhật phiên bản mới để loại bỏ lỗ hổng bảo mật.

### **1.1.5. Xu hướng và thách thức trong an ninh mạng**

Sự phát triển của công nghệ kéo theo nhiều thách thức mới trong lĩnh vực an ninh mạng, bao gồm:

* Trí tuệ nhân tạo (AI): Cả hai bên (tấn công và phòng thủ) đều đang sử dụng AI để tăng cường hiệu quả.
* Internet of Things (IoT): Số lượng thiết bị kết nối ngày càng nhiều khiến phạm vi bị tấn công ngày càng rộng.
* Tấn công không gian mạng (Cyberwarfare): Các quốc gia sử dụng chiến tranh mạng như một công cụ chiến lược.
* Thiếu nhân lực an ninh mạng: Sự thiếu hụt chuyên gia khiến việc bảo vệ mạng trở nên khó khăn hơn.

### **1.1.6. Kết luận**

An ninh mạng là một lĩnh vực không ngừng phát triển và đòi hỏi sự đầu tư lớn về cả nguồn lực con người lẫn công nghệ. Việc nâng cao nhận thức và trang bị kiến thức về an ninh mạng không chỉ giúp bảo vệ hệ thống khỏi các mối đe dọa mà còn đảm bảo sự phát triển bền vững trong môi trường số hóa ngày nay.

## **1.2 Tổng quan về thuật toán DES**

### **1.2.1 Giới thiệu chung về thuật toán DES**

DES (Data Encryption Standard) là chuẩn mã hóa dữ liệu đầu tiên trên thế giới, do Cơ quan an ninh Quốc gia Hoa Kỳ (NSA) đề xuất trên cơ sở cải tiến thuật toán Lucifer do hãng IBM công bố năm 1964. DES đã được sử dụng rộng rãi ở Hoa Kỳ và nhiều quốc gia khác trong các thập kỷ 70, 80, 90 cho đến khi được thay thế bởi Tiêu chuẩn mã hóa dữ liệu tiên tiến AES (Advanced Encryption Standard) vào năm 2002.

A diagram of a computer flowchart

Description automatically generated

Hình 1:

Đầu vào của DES là khối 64 bit, đầu ra cũng là khối 64 bit. Khóa mã hóa có độ dài 56 bit, nhưng thực chất ban đầu là 64 bit, được lấy đi các bit ở vị trí chia hết cho 8 dùng để kiểm tra tính chẵn lẻ.

### **1.2.2 Thuật toán**

DES là thuật toán mã hóa theo khối, nó xử lý từng khối thông tin của bản rõ có độ dài xác định là 64 bit. Trước khi đi vào 16 chu trình chính, khối dữ liệu cần bảo mật sẽ được tách ra thành từng khối 64 bit, và từng khối 64 bit này sẽ lần lượt được đưa vào 16 vòng mã hóa DES để thực hiện. **Input:** Bản rõ M = m1m2…m64 là một khối 64 bit, khóa 64 bit K = k1k2…k64. **Output:** Bản mã 64 bit C = c1c2… c64

Bước 1: Sinh khóa con: Sử dụng thuật toán sinh khóa con từ khóa K ta sẽ được 16 khóa con K1, K2, … K16

Bước 2: Sử dụng phép hoán vị khởi đầu IP (Initial Permutation) để hoán vị các bit của M, kết quả nhận được chia thành 2 nửa là L0 = m63m62…m32, R0 = m31m30…m0.

Bước 3: Với i chạy từ i = 1 đến 16 thực hiện: Tính các Li và Ri theo công thức: Li = Ri-1 Ri = Li-1 XOR f(Ri-1, Ki) trong đó f(Ri-1, Ki) = R(S(E(Ri-1) XOR Ki)). Việc tính f(Ri-1, Ki) sẽ được trình bày chi tiết ở phần sau.

Bước 4: Đổi vị trí khối L16, R16 ta được khối R16L16 = b1b2…b64.

Bước 5: Sử dụng phép hoán vị kết thúc FP(Final Permutation – nghịch đảo với hoán vị khởi đầu IP) ta thu được bản mã cần tìm : C = IP-1(b1b2…b64)

A diagram of a computer

Description automatically generated

**Hình 2:**

### **1.2.3 Quá trình sinh khóa con**

16 vòng lặp của DES chạy cùng thuật toán như nhau nhưng với 16 khóa con khác nhau. Các khóa con đều được sinh ra từ khóa chính của DES bằng thuật toán sinh khóa con.

A diagram of a computer flowchart

Description automatically generated

Hình 3:

Khóa ban đầu là 1 xâu có độ dài 64 bit, bit thứ 8 của mỗi byte sẽ được lấy ra để kiểm tra phát hiện lỗi, tạo ra chuỗi 56 bit. Sau khi bỏ các bit kiểm tra ta sẽ hoán vị chuỗi 56 bit này. Hai bước trên được thực hiện thông qua hoán vị ma trận PC-1 (Permuted choice 1).

A number grid with black numbers

Description automatically generated

Hình 4:

Tiếp theo ta kết quả sau khi PC-1 thành 2 phần : C0 : 28 bit đầu. D0 : 28 bit cuối. Mỗi phần sẽ được xử lý 1 cách độc lập. Ci = LSi(Ci-1) Di = LSi(Ci-1) với 1 ≤ i ≤ 16. LSi là biểu diễn phép dịch bit vòng (cyclic shift) sang trái 1 hoặc 2 vị trí tùy thuộc vào i.

| **Vòng lặp** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Số lần dịch trái | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 |

Cuối cùng sử dụng hoán vị cố định PC-2 (Permuted choice 2) để hoán vị chuỗi CiDi 56 bit tạo thành khóa Ki với 48 bit.

A number grid with numbers

Description automatically generated

Hình 5:

### **1.2.4 Quá trình mã hóa DES**

Chia thành 3 giai đoạn:

**Giai đoạn 1:**

Với bản rõ cho trước x, 1 xâu x’ sẽ được tạo ra bằng cách hoán vị các bit của x theo hoán vị ban đầu IP.

Tiếp theo x’ sẽ được chia thành 2 phần L0,R0. x’ = IP(x) = L0R0 Trong đó L0 là 32 bit đầu, R0 là 32 bit cuối.

A grid of numbers with black text

Description automatically generated

Hình 6:

**Giai đoạn 2:**

Tính toán 16 lần bằng 1 hàm xác định. Ta sẽ tính Li, Ri (1 ≤ i ≤ 16) theo quy tắc: Li = Ri-1. Ri = Li-1 XOR f(Ri-1, Ki). Với Ki là khóa được sinh ra ở quá trình tạo khóa, f là một hàm sẽ được trình bày ở phần sau.

A diagram of a algorithm

Description automatically generated

Hình 7:

**Giai đoạn 3:**

Áp dụng hoán vị kết thúc FP cho xâu bit R16L16 ta thu được bản mã y: y = FP(R16L16).

A grid of numbers with black text

Description automatically generated

Hình 8:

### **1.2.5 Giải mã DES**

Quá trình giải mã của DES cũng tương tự quá trình mã hóa. Chỉ khác nhau ở: Li = Ri-1. Ri = Li-1 XOR f(Ri-1, K16-i+1). Như vậy khóa K của hàm F sẽ đi từ khóa K16 đến khóa K1.

A diagram of a computer

Description automatically generated

Hình 9:

## **1.3 Tổng quan về ngôn ngữ lập trình java**

### **1.3.1 Khái niệm**

Java là một ngôn ngữ lập trình bậc cao, hướng đối tượng, được phát triển bởi Sun Microsystems vào năm 1995 và hiện thuộc sở hữu của Oracle Corporation. Ngôn ngữ này nổi bật với khả năng "Viết một lần, chạy ở mọi nơi" (Write Once, Run Anywhere - WORA), cho phép các ứng dụng Java chạy trên nhiều nền tảng mà không cần biên dịch lại.

### **1.3.2 Lịch sử và phát triển**

Java được khởi xướng bởi James Gosling và nhóm kỹ sư tại Sun Microsystems với mục tiêu ban đầu là phát triển một ngôn ngữ cho các thiết bị điện gia dụng. Ban đầu, nó được gọi là Oak, trước khi đổi tên thành Java. Kể từ khi ra mắt, Java đã trải qua nhiều phiên bản và cải tiến, giữ vững vị thế là một trong những ngôn ngữ lập trình phổ biến nhất thế giới.

### **1.3.3 Đặc điểm nổi bật của Java**

Hướng đối tượng: Java hoàn toàn dựa trên mô hình lập trình hướng đối tượng, cho phép tổ chức mã nguồn theo các đối tượng và lớp. Điều này giúp tăng cường tính tái sử dụng mã và khả năng bảo trì.

Độc lập nền tảng: Mã Java được biên dịch thành bytecode, có thể chạy trên bất kỳ máy ảo Java (JVM) nào. Điều này giúp ứng dụng Java có thể hoạt động trên nhiều hệ điều hành khác nhau mà không cần thay đổi mã nguồn.

Bảo mật: Java được thiết kế với nhiều tính năng bảo mật mạnh mẽ như kiểm tra kiểu tĩnh và khả năng quản lý bộ nhớ an toàn. Điều này giúp giảm thiểu các lỗ hổng bảo mật trong ứng dụng.

Đơn giản và dễ học: Cú pháp của Java tương tự như C và C++, nhưng loại bỏ nhiều tính năng phức tạp như con trỏ, làm cho việc học trở nên dễ dàng hơn cho người mới bắt đầu.

Đa luồng: Hỗ trợ lập trình đa luồng, cho phép thực hiện nhiều tác vụ đồng thời, điều này rất hữu ích trong phát triển ứng dụng tương tác và game.

### **1.3.4 Kiến trúc của Java**

Java có ba thành phần chính trong kiến trúc của nó:

* Java Virtual Machine (JVM): Là môi trường thực thi cho các chương trình Java. JVM chịu trách nhiệm biên dịch bytecode thành mã máy cụ thể cho hệ điều hành mà nó đang chạy.
* Java Runtime Environment (JRE): Cung cấp môi trường cần thiết để chạy các ứng dụng Java, bao gồm JVM và các thư viện cần thiết.
* Java Development Kit (JDK): Là bộ công cụ phát triển bao gồm JRE cùng với các công cụ khác như biên dịch viên (compiler) để xây dựng ứng dụng Java.

### **1.3.5 Ứng dụng của Java**

Java được sử dụng rộng rãi trong nhiều lĩnh vực:

* Phát triển ứng dụng web: Nhiều hệ thống quản lý nội dung (CMS), thương mại điện tử và dịch vụ trực tuyến đều được xây dựng bằng Java.
* Phát triển ứng dụng di động: Là ngôn ngữ chính để phát triển ứng dụng Android, chiếm ưu thế trong thị trường di động.
* Phát triển phần mềm doanh nghiệp: Nhiều giải pháp phần mềm lớn cho doanh nghiệp sử dụng Java nhờ vào tính ổn định và khả năng mở rộng.
* Hệ thống nhúng: Được sử dụng trong các thiết bị nhúng nhờ vào tính linh hoạt và khả năng tương thích cao.

### **1.3.6 Tính năng nổi bật của Java**

* Mạnh mẽ: Java cung cấp cơ chế kiểm tra lỗi tại thời điểm biên dịch và runtime, giúp giảm thiểu lỗi trong ứng dụng.
* Portable: Các ứng dụng Java có thể chạy trên bất kỳ nền tảng nào hỗ trợ JVM mà không cần thay đổi mã nguồn.
* Hiệu suất cao: Mặc dù không nhanh bằng các ngôn ngữ biên dịch như C++, nhưng với tối ưu hóa JVM, hiệu suất của Java vẫn rất ấn tượng.
* Quản lý bộ nhớ tự động: Java sử dụng garbage collection để quản lý bộ nhớ hiệu quả, giúp giảm thiểu rò rỉ bộ nhớ.

## **1.4 Tổng quan về ngôn ngữ lập trình C#**

### **1.4.1 Khái niệm**

C# (C Sharp) là một ngôn ngữ lập trình hiện đại do Microsoft phát triển, được giới thiệu lần đầu vào năm 2000. Ngôn ngữ này được thiết kế chủ yếu để phát triển ứng dụng trên nền tảng .NET, mang lại nhiều tính năng mạnh mẽ cho lập trình viên.

### **1.4.2 Lịch sử phát triển**

C# đã trải qua nhiều phiên bản và cải tiến từ khi ra mắt. Các phiên bản chính bao gồm:

C# 1.0: Phiên bản đầu tiên với các tính năng cơ bản của lập trình hướng đối tượng.

C# 2.0: Giới thiệu kiểu dữ liệu generic, nullable types và anonymous methods, giúp mã nguồn trở nên sạch sẽ và dễ bảo trì hơn 2.

C# 3.0: Cải tiến đáng kể với sự ra đời của LINQ (Language Integrated Query), lambda expressions và extension methods, cho phép truy vấn dữ liệu một cách tự nhiên và dễ dàng hơn 23.

C# 5.0: Thêm hỗ trợ cho lập trình bất đồng bộ với từ khóa async và await, giúp cải thiện hiệu suất cho các ứng dụng cần xử lý nhiều tác vụ đồng thời 3.

C# 9.0 và 10.0: Cải tiến tính năng với các kiểu dữ liệu mới như record types, giúp dễ dàng quản lý và xử lý dữ liệu không thay đổi 3.

### **1.4.3 Đặc điểm nổi bật**

Hướng đối tượng: C# hoàn toàn hỗ trợ lập trình hướng đối tượng với các khái niệm như lớp, đối tượng, kế thừa, và đa hình. Điều này giúp tổ chức mã nguồn một cách rõ ràng và dễ bảo trì.

Quản lý bộ nhớ tự động: Sử dụng Garbage Collector để tự động quản lý bộ nhớ, giảm thiểu rủi ro rò rỉ bộ nhớ mà không cần can thiệp thủ công từ lập trình viên.

Kiểm soát kiểu dữ liệu chặt chẽ: C# yêu cầu khai báo kiểu dữ liệu rõ ràng, giúp phát hiện lỗi sớm trong quá trình biên dịch.

Tính năng mở rộng: Với extension methods, lập trình viên có thể thêm các phương thức mới cho các lớp mà không cần thay đổi mã nguồn gốc 1.

### **1.4.4 Ứng dụng của C#**

C# được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau:

Phát triển ứng dụng desktop: Thông qua Windows Forms và WPF (Windows Presentation Foundation), C# cho phép xây dựng các ứng dụng giao diện người dùng phong phú.

Phát triển web: ASP.NET là framework mạnh mẽ cho việc xây dựng ứng dụng web động với khả năng mở rộng cao.

Phát triển game: Unity là một trong những công cụ phát triển game phổ biến nhất sử dụng C#, cho phép tạo ra các trò chơi đa nền tảng 5.

Ứng dụng di động: Xamarin cho phép phát triển ứng dụng di động trên cả iOS và Android bằng cách sử dụng C#.

### **1.4.5 Ưu điểm của C#**

Dễ học và dễ sử dụng: Với cú pháp rõ ràng, C# là lựa chọn lý tưởng cho cả người mới bắt đầu lẫn lập trình viên có kinh nghiệm.

Tính năng phong phú: Hỗ trợ nhiều tính năng hiện đại như async/await, LINQ, generics giúp viết mã nhanh chóng và hiệu quả.

Cộng đồng lớn mạnh: Nhiều tài liệu học tập, khóa học trực tuyến và diễn đàn hỗ trợ người dùng trong quá trình học tập và phát triển 5.

# **Chương 2: Ứng dụng của DES trong mã hóa dữ liệu**

* 1. Nghiên cứu, tìm hiểu về mô hình mạng an toàn (Các thành phần chính của mô hình mạng an toàn, các công nghệ hỗ trợ)
  2. Các mô hình mạng an toàn sử dụng trong doanh nghiệp
  3. Nghiên cứu mô hình mạng an toàn cho công ty….

Trình bày mô hình cụ thể: cấu trúc cơ bản, các thành phần chính, cách thức hoạt động, ứng dụng…

* 1. Thực nghiệm mô hình mạng an toàn cho công ty …
* Giới thiệu công cụ hỗ trợ mô phỏng: packet tracer,…
* Mục tiêu của hệ thống thực nghiệm
* Thiết kế kịch bản chương trình
* Giải pháp bảo mật trong hệ thống thực nghiệm
* Mô phỏng, triển khai thực nghiệm

# **Chương 3: Kết luận và bài học kinh nghiệm**

## **3.1 Kiến thức kỹ năng đã học được trong quá trình thực hiện đề tài.**

## **3.2 Bài học kinh nghiệm**

**3.3 Đề xuất về tính khả thi của chủ đề nghiên cứu, những thuận lợi, khó khăn …**

**Tài liệu tham khảo**

(Sắp xếp theo ngôn ngữ: Việt, Anh, Pháp, Đức, Nga…)

[1]. Nguyễn Xuân Dũng, Bảo mật thông tin – Mô Hình và ứng dụng, NXB thống kê, 2009.

[2]. Bùi Doãn Khanh, Nguyễn Đình Thúc, Mã hóa thông tin – Lý thuyết và ứng dụng, NXB Lao động xã hội, 2011.

[3]. William Stallings**,** Cryptography and Network Security Principles and Practices**,** Fourth Edition, Prentice Hall, 2005.

…

[n]. <http://www.hhs.com>

(thời gian truy cập: 15h00, ngày 27/10/2024)

**Phụ lục 1: Chương trình Java**

**Phụ lục 2: Chương trình C#**