**TRƯỜNG ĐẠI HỌC KHOA HỌC**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\*\*\***

Ảnh có chứa văn bản, biểu tượng, Nhãn hiệu, Xanh điện

Mô tả được tạo tự động

**ĐỀ TÀI:**

**TÌM HIỂU VÀ TRIỂN KHAI GIẢI PHÁP OTP**

|  |  |  |
| --- | --- | --- |
| Tên sv | : | Nguyễn Văn Đình |
| Mã sv | : | 21T1020302 |
| Môn học | : | An ninh mạng - Nhóm 8 |
| Giáo viên hướng dẫn | : | Nguyễn Quang Hưng |

Huế 2024

**DANH MỤC BẢNG VÀ HÌNH VẼ**

Hình 2.1. Xác thực bằng mật khẩu một lần theo Lamport 20

Hình 2.2. Tấn công MITD lên giao thức xác thực sử dụng OTP theo Lamport 23

Hình 2.3. Thiết bị phần cứng Token sinh mật khẩu OTP 26

Hình 2.4. Chia sẻ giá trị bí mật khi sinh OTP ở cả hai bên xác thực27

Hình 2.5. Sơ đồ mô tả thuật toán HOTP sinh mật khẩu OTP 27

Hình 2.6. Sơ đồ tổng thể của xác thực với OTP được sinh ở phía verifier 35

Hình 2.7. Mô hình chức năng của một DRBG 37

Hình 2.8. Tin nhắn SMS cung cấp mật khẩu OTP 42

Hình 2.9. Email cung cấp mật khẩu OTP 43

Hình 2.10. Lựa chọn dịch vụ của ngân hàng để giao dịch 46

Hình 2.11. Xác nhận giao dịch ngân hàng 47

Hình 2.12. Website Thương mại điện tử 49

Hình 2.13. Cổng thanh toán sử dụng xác thực với OTP 55

Hình 3.2. Quá trình sinh mã OTP 56

[Hình 3.3. Quá trình xác thực mã OTP 56](file://localhost/C:/Users/MAYMOT/Desktop/luan%20van%20chinh%20sua%20theo%20yeu%20cau%20cua%20thay_LONGNGNGNG.docx%23_Toc452027503)

Hình 3.4. Mô hình sử dụng OTP để xác thực tài khoản trong hệ thông học trực tuyến 57

**Danh mục viết tắt:**

|  |  |
| --- | --- |
| **Ký hiệu** | **Ý nghĩa của ký hiệu** |
| OTP | One Time Password |
| PIN | Personal Identification Number |
| DRBG | Deterministic Random Bit Generator |
| PRNG | Pseudorandom Number Generator |
| HOTP | HMAC-Based One-Time Password Algorithm |
| TOTP | Time - Based One Time Password |
| HMAC | Hash message authentication code |
| MITD | Man-in-the-middle |
| Claimant | Bên yêu cầu xác thực |
| Verifier | Bên xác thực |
| C | Counter |
| K | Key |
| T | Time |
| CSDL | Cơ sở dữ liệu |

**Mục lục**

[**MỞ ĐẦU** 5](#_Toc186805185)

[1.  Lý do chọn đề tài 5](#_Toc186805186)

[2.  Đối tượng và phạm vi nghiên cứu 7](#_Toc186805187)

[3.  Hướng nghiên cứu của đề tài 7](#_Toc186805188)

[4.  Những nội dung nghiên cứu chính 7](#_Toc186805189)

[**CHƯƠNG 1. TỔNG QUAN VỀ XÁC THỰC** 8](#_Toc186805190)

[1.  Khái niệm về xác thực 8](#_Toc186805191)

[2.  Các yếu tố xác thực 9](#_Toc186805192)

[3.  Một số phương pháp xác thực 10](#_Toc186805193)

[**CHƯƠNG 2: PHƯƠNG PHÁP XÁC THỰC SỬ DỤNG MẬT KHẨU MỘT LẦN (OTP)** 15](#_Toc186805194)

[1. Giới thiệu về mật khẩu OTP 15](#_Toc186805195)

[2. Xác thực với OTP được sinh ở phía claimant 19](#_Toc186805196)

[3. Xác thực với OTP được sinh ở cả hai phía 23](#_Toc186805197)

[4  Xác thực với OTP được sinh ở phía verifier 33](#_Toc186805198)

[5  Một số ứng dụng OTP trong thực tế hiện nay 44](#_Toc186805199)

[**CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH ỨNG DỤNG MẬT KHẨU OTP CHO HỆ THỐNG HỌC TẬP** 51](#_Toc186805200)

[1.  Đặt vấn đề 51](#_Toc186805201)

[2. Ứng dụng mật khẩu OTP cho hệ thống học tập trực tuyến. 51](#_Toc186805202)

[3. Xây dựng chương trình ứng dụng sinh OTP cho hoạt động học tập trực tuyến. 55](#_Toc186805203)

[**KẾT LUẬN** 62](#_Toc186805204)

[1.  Kết luận 62](#_Toc186805205)

[2.  Kiến nghị 62](#_Toc186805206)

[3.  Hướng phát triển của đề tài 63](#_Toc186805207)

[**TÀI LIỆU THAM KHẢO** 63](#_Toc186805208)

[Tài liệu tiếng việt 63](#_Toc186805209)

[Tài liệu tiếng anh 64](#_Toc186805210)

# **MỞ ĐẦU**

## 1.  Lý do chọn đề tài

Hiện nay, khi mạng Internet đã được mở rộng thì nhu cầu sử dụng các dịch vụ, trao đổi thông tin, dữ liệu của người dùng qua mạng cũng tăng lên, và trở thành vấn đề cần được quan tâm và đáp ứng. Tuy nhiên, bên cạnh những lợi ích mà Internet mang lại cũng tiềm ẩn những hiểm họa và nguy cơ mất an toàn như tài khoản ngân hàng của cá nhân, tài khoản thanh toán giao dịch trực tuyến,...có thể bị đánh cắp và sử dụng vào mục đích bất hợp pháp. Chính vì vậy, có rất nhiều giải pháp nhằm đảm bảo an toàn cho thông tin của người sử dụng được triển khai và thiết lập, như quản lý tài khoản người dùng thông qua Username và Password, xác thực khi truy cập sử dụng dịch vụ, một số giải pháp bảo vệ hệ thống Web như sử dụng hệ thống phát hiện xâm nhập, tường lửa để ngăn chặn và cảnh báo truy cập trái phép vào hệ thống, các công cụ quét và phát hiện mã độc hại tiềm ẩn trên các Website,... trong đó thì phương pháp xác thực người dùng khi sử dụng một dịch vụ trên mạng là một điều quan trọng, nó giúp đảm bảo an toàn cho hệ thống cũng như đảm bảo thông tin riêng cho người dùng.

Đối với phương pháp xác thực phổ biến là sử dụng tài khoản đăng nhập của người dùng gồm Username và Password đã bộc lộ những điểm yếu khi có thể bị đánh cắp trong trường hợp máy tính của người dùng bị cài đặt các phần mềm, chương trình có khả năng ăn cắp dữ liệu, hay trong quá trình trao đổi thông tin qua mạng bị nghe lén trên đường truyền, bị chuyển hướng đến trang Web chứa mã độc hại và bị lừa đảo chiếm mất tài khoản. Để đảm bảo an toàn hơn cho người dùng, tránh được các nguy cơ mất an toàn thông tin đó, các nhà cung cấp dịch vụ đã áp dụng những phương pháp xác thực mạnh hơn khi kết hợp các yếu tố có được từ người dùng như Username, Password, mã PIN, đặc điểm sinh trắc học,...Trong đó, phương pháp xác thực hai nhân tố với mật khẩu sử dụng một lần (OTP) kết hợp thông tin về Username, Password của người dùng hiện đang được áp dụng phổ biến trong các giao dịch ngân hàng trực tuyến, mua sắm, thanh toán giao dịch trực tuyến trên các Website thương mại điện tử.

Mật khẩu OTP được áp dụng giúp nâng cao an toàn cho người dùng và cho hệ thống trong quá trình xác thực trước khi sử dụng và cung cấp các dịch vụ. Mật khẩu OTP rất linh hoạt, thuận tiện cho người dùng khi nó có thể được gửi thông qua tin nhắn SMS tới số điện thoại di động, Email của người dùng, hay thông qua các thiết bị phần cứng như Token sinh OTP trong quá trình xác thực các giao dịch quan trọng như ngân hàng, thanh toán trực tuyến.

Trong quá trình thực hiện làm luận văn tốt nghiệp tôi thực hiện nghiên cứu đề tài *"Nghiên cứu, tìm hiểu phương pháp xác thực dùng mật khẩu sử dụng một lần (OTP) và ứng dụng trong giao dịch trực tuyến"*, đây là đề tài có ý nghĩa thiết thực khi nghiên cứu về các cơ chế sinh mật khẩu OTP, các phương pháp phân phối, ứng dụng của mật khẩu OTP trong thực tế hiện nay. Thông qua đó có thể hiểu rõ hơn lý do vì sao mật khẩu OTP được áp dụng là giải pháp nhằm nâng cao độ an toàn, tin cậy trong quá trình xác thực người dùng khi truy cập, sử dụng các dịch vụ, tài nguyên và các giao dịch trực tuyến.

## 2.  Đối tượng và phạm vi nghiên cứu

a.  Đối tượng:

-  Tập trung nghiên cứu đối tượng mật khẩu sử dụng một lần OTP.

b.  Phạm vi:

-  Nghiên cứu và ứng dụng mật khẩu sử dụng một lần OTP trong các giao dịch trực tuyến.

## 3.  Hướng nghiên cứu của đề tài

-  Tìm hiểu một số phương pháp xác thực.

-  Nghiên cứu về xác thực sử dụng mật khẩu OTP.

-  Tìm hiểu một số ứng dụng mật khẩu OTP trong thực tế.

-  Cài đặt và thử nghiệm.

## 4.  Những nội dung nghiên cứu chính

Luận văn được trình bày trong 3 chương. Các nội dung cơ bản của luận văn được trình bày theo cấu trúc như sau:

Chương 1: Tổng quan về xác thực. Trình bày các khái niệm cơ bản về xác thực, nhân tố xác thực, và một số phương pháp xác thực sử dụng hiện nay.

Chương 2: Phương pháp xác thực sử dụng mật khẩu OTP. Trình bày khái niệm về OTP, các cơ chế sử dụng mật khẩu OTP, và đưa ra một số thuật toán sinh mật khẩu OTP.

Chương 3: Ứng dụng OTP trong học tập trực tuyến. Trình bày mô hình hoạt động của học trực tuyến, kỹ thuật sinh mật khẩu OTP, các kết quả thử nghiệm của hệ thống đăng nhập trực tuyến có sử dụng mật khẩu OTP.

# **CHƯƠNG 1. TỔNG QUAN VỀ XÁC THỰC**

## 1.  Khái niệm về xác thực

Xác thực (Authentication) là việc xác lập hoặc chứng thực một thực thể (người nào đó hay một cái gì đó) đáng tin cậy, có nghĩa là những thông tin do một người đưa ra hoặc về một cái gì đó là đúng đắn. Xác thực một đối tượng còn có nghĩa là công nhận nguồn gốc của đối tượng, còn xác thực một người thường bao gồm việc thẩm tra nhận dạng cá nhân của họ. Việc xác thực thường phụ thuộc vào một hoặc nhiều yếu tố xác thực (authentication factor) làm minh chứng cụ thể.

Xác thực là khâu đặc biệt quan trọng để bảo đảm an toàn cho hoạt động của một hệ thống thông tin. Đó là một quy trình nhằm xác minh nhận dạng số (digital identity) của bên gửi thông tin (sender) trong liên lạc trao đổi, xử lý thông tin, chẳng hạn như một yêu cầu đăng nhập. Bên gửi cần phải xác thực có thể là một người sử dụng máy tính, bản thân một máy tính hoặc một phần mềm. Đầu tiên, hệ thống luôn xác thực một thực thể khi nó cố gắng thử thiết lập liên lạc. Khi đó, nét nhận dạng của thực thể được dùng để xác định sự truy nhập của thực thể đó như một đặc quyền hoặc để đạt được sự sẵn sàng phục vụ. Đối với các giao dịch ngân hàng điện tử điển hình như giao dịch qua ATM/POS, giao dịch Online/Internet Banking, giao dịch Mobile Banking... thì xác thực là bắt buộc trong quản lý truy cập

## 2.  Các yếu tố xác thực

Những yếu tố xác thực cho người sử dụng có thể được phân loại như sau:

* Những cái mà người sử dụng sở hữu bẩm sinh, chẳng hạn như dấu vân tay hoặc mẫu dạng võng mạc mắt, chuỗi ADN, mẫu dạng giọng nói, chữ ký, tín hiệu sinh điện đặc thù do cơ thể sống tạo ra, hoặc những định dạng sinh trắc học khác.
* Những cái người sử dụng có, chẳng hạn như chứng minh thư, chứng chỉ an ninh (security token), chứng chỉ phần mềm (software token) hoặc điện thoại di động....
* Những gì người sử dụng biết, chẳng hạn như mật khẩu (password), mật ngữ (pass phrase) hoặc mã số định danh cá nhân (personal identification number - PIN)....

Trong thực tế, nhiều khi một tổ hợp của những yếu tố trên được sử dụng, lúc đó người ta nói đến xác thực đa yếu tố. Chẳng hạn trong giao dịch ATM, thẻ ngân hàng và mã số định danh cá nhân (PIN) được sử dụng – trong trường hợp này là một trong các dạng xác thực hai yếu tố (two – factor authentication – 2FA).

## 3.  Một số phương pháp xác thực

Hiện nay, trong các giao dịch trực tuyến, một số phương pháp xác thực phổ biến gồm:

**3.1  Xác thực dựa trên định danh người sử dụng (Username) và mật khẩu (Password)**

Sự kết hợp của một cặp Username và Password là cách xác thực phổ biến nhất hiện nay. Với phương thức xác thực này, thông tin cặp username và password nhập vào được đối chiếu với dữ liệu đã được lưu trữ trên hệ thống. Nếu thông tin trùng khớp thì người sử dụng được xác thực, còn nếu không người sử dụng bị từ chối hoặc cấm truy cập.

Phương thức xác thực này có tính bảo mật không cao, vì thông tin cặp Username và Password dùng đăng nhập vào hệ thống mà ta gửi đi xác thực là trong tình trạng ký tự văn bản rõ, tức không được mã hóa và có thể bị chặn bắt trên đường truyền, thậm chí ngay trong quá trình nhập vào Password còn có thể bị lộ do đặt quá đơn giản (dạng ‘123456’, ‘abc123’ v.v.) hoặc dễ đoán (tên, ngày sinh của người thân...).

**3.2. Sử dụng giao thức bắt tay có thử thách (Challenge Handshake Authentication Protocol – CHAP)**

Đây cũng là mô hình xác thực dựa trên username/password. Khi người dùng (User) thực hiện thủ tục đăng nhập (log on), máy chủ (server) đảm nhiệm vai trò xác thực sẽ gửi một thông điệp thử thách (challenge message) cho máy tính của người dùng. Lúc này máy tính của người dùng sẽ phản hồi lại bằng Username và password được mã hóa. Máy chủ xác thực sẽ so sánh phiên bản xác thực người dùng được lưu giữ với phiên bản mã hóa vừa nhận, nếu trùng khớp thì người dùng sẽ được xác thực. Để đảm bảo an toàn, bản thân password không bao giờ được gửi qua mạng.

Phương thức CHAP thường được sử dụng khi người dùng đăng nhập vào các máy chủ ở xa (remote server) của hệ thống, chẳng hạn như RAS server. Dữ liệu chứa password được mã hóa đôi khi được gọi là “mật khẩu băm” (hash password) theo tên của phương pháp mã hoá dùng các hàm băm.

**3.3. Xác thực Kerberos**

Là nền tảng xác thực chính của nhiều hệ điều hành như UNIX, Windows Xác thực Kerberos dùng một máy chủ trung tâm để kiểm tra việc xác thực người dùng và cấp phát thẻ dịch vụ (service ticket) để người dùng có thể truy cập vào tài nguyên hệ thống. Xác thực Kerberos là một phương thức có tính an toàn cao nhờ việc dùng thuật toán mã hóa mạnh. Kerberos cũng dựa trên độ chính xác của thời gian xác thực giữa máy chủ và người dùng, do đó cần phải đảm bảo kết nối đồng bộ thời gian giữa các thành phần này của hệ thống.

**3.4  Xác thực sử dụng token**

Token là những phương tiện vật lý như các thẻ thông minh (smart card), thẻ đeo của nhân viên (ID badge) chứa thông tin xác thực hoặc bộ tạo mật khẩu dùng một lần (One Time Password - OTP).

OTP là mật khẩu dùng một lần, được tạo ra trên bộ tạo OTP (token) và kiểm tra trong hệ thống bảo mật riêng. OTP tự động thay đổi thường xuyên và chỉ tồn tại trong một thời gian ngắn (khoảng vài chục giây) cho từng lần truy nhập.

Token có thể lưu trữ mã số nhận dạng cá nhân (PIN), thông tin về người dùng, lưu giữ hoặc tạo ra password. Các thông tin trên token chỉ có thể được đọc/xử lý bởi các thiết bị hoặc hệ thống đặc dụng. Chẳng hạn như thẻ thông minh được đọc bởi đầu đọc thẻ smart card chuyên dụng, OTP được xử lý bởi hệ thống xác thực sử dụng yếu tố xác thực thứ hai là mật khẩu dùng một lần.

**Ví dụ về Smart Cards**

Smart cards là ví dụ điển hình về xác thực token. Một smart card là một thẻ nhựa có gắn một chip máy tính lưu trữ các loại thông tin điện tử khác nhau. Nội dung thông tin của card được đọc với một thiết bị đặc biệt.

**3.5 Xác thực áp dụng các phương pháp nhận dạng sinh trắc học (Biometrics)**

Đây là mô hình xác thực có tính bảo mật cao dựa trên đặc điểm sinh học của từng cá nhân, trong đó sử dụng các thủ tục như quét dấu vân tay (fingerprint scanner), quét võng mạc mắt (retinal scanner), nhận dạng giọng nói (voice - recognition), nhận dạng khuôn mặt (face recognition)  Nhờ các tiến bộ vượt bậc của công nghệ sinh học, phương thức xác thực dựa trên nhận dạng sinh trắc học ngày càng trở nên phổ biến và được chấp nhận rộng rãi.

**3.6 Phương thức xác thực lẫn nhau (Mutual Authentication)**

Đây là phương thức bảo mật trong đó các thành phần tham gia giao tiếp với nhau sẽ kiểm tra, xác thực lẫn nhau. Chẳng hạn, trong một hệ thống mạng Client/Server, trước hết máy chủ (chứa tài nguyên) kiểm tra “giấy phép truy cập” của người dùng và sau đó người dùng lại kiểm tra “giấy phép cấp tài nguyên” của máy chủ. Cũng tương tự như vậy, khi khách hàng thực hiện giao dịch với hệ thống E-Banking của một Ngân hàng đã chọn, thì cần phải kiểm tra xem hệ thống đó có đúng là của Ngân hàng đó không và ngược lại hệ thống E- Banking của Ngân hàng cũng kiểm tra chính khách hàng thực hiện giao dịch.

**3.7  Xác thực đa yếu tố**

Xác thực đa yếu tố (Multi-Factor Authentication) là phương thức xác thực dựa trên nhiều yếu tố xác thực kết hợp, là mô hình xác thực yêu cầu kiểm chứng ít nhất là hai yếu tố xác thực. Phương thức này là sự kết hợp của bất cứ yếu tố xác thực nào, ví dụ như yếu tố đặc tính sinh trắc của người dùng hoặc những gì người dùng biết để xác thực trong hệ thống.

Với xác thực đa yếu tố, ngân hàng có thể tăng mức độ an toàn, bảo mật cho giao dịch trực tuyến lên rất nhiều nhờ việc kiểm chứng nhiều yếu tố xác thực. Ví dụ như xác thực chủ thẻ trong giao dịch ATM, yếu tố xác thực đầu tiên của khách hàng là thẻ ATM (cái khách hàng có), sau khi đưa thẻ vào máy, khách hàng sẽ phải đưa tiếp yếu tố xác thực thứ hai là số PIN (cái khách hàng biết). Một ví dụ khác là xác thực người sử dụng dịch vụ giao dịch Internet Banking: khách hàng đăng nhập với Username và Password sau đó còn phải cung cấp tiếp OTP (One - Time - Password) được sinh ra trên token của riêng khách hàng.

An toàn, bảo mật trong giao dịch trực tuyến là hết sức quan trọng, trong đó xác thực người sử dụng là một trong những khâu cốt lõi. Với xác thực đa yếu tố, ta có thể tăng mức độ an toàn, bảo mật nhờ việc kiểm chứng nhiều yếu tố xác thực. Mức độ an toàn bảo mật sẽ càng cao khi số yếu tố xác thực càng nhiều. Khi số yếu tố xác thực lớn thì hệ thống càng phức tạp, kéo theo chi phí đầu tư và duy trì vận hành tốn kém, đồng thời lại bất tiện cho người sử dụng. Do vậy, trên thực tế để cân bằng giữa an toàn, bảo mật và tính tiện dụng, người ta thường áp dụng xác thực hai yếu tố và xác thực ba yếu tố (three-factor authentication - 3FA).

Xác thực đa yếu tố dù có mức độ an toàn, bảo mật cao hơn, nhưng cũng cần các biện pháp nghiệp vụ khác để bảo đảm tuyệt đối an toàn trong các hoạt động giao dịch trực tuyến.

# **CHƯƠNG 2: PHƯƠNG PHÁP XÁC THỰC SỬ DỤNG MẬT KHẨU MỘT LẦN (OTP)**

## 1. Giới thiệu về mật khẩu OTP

Mật khẩu sử dụng một lần (OTP) hiện nay đang được sử dụng để xác thực trong các giao dịch ngân hàng, giao dịch trực tuyến,...Và mật khẩu OTP có những ưu điểm giúp nâng cao an toàn cho người dùng trong quá trình xác thực nhằm tránh được các rủi ro mất an toàn thông tin.

**1.1  Khái niệm mật khẩu OTP**

Mật khẩu sử dụng một lần (OTP) là loại mật khẩu chỉ dùng một lần và chỉ có giá trị cho một phiên đăng nhập hoặc cho một giao dịch trong một khoảng thời gian nhất định. OTP thường được tạo ra dựa trên các thông tin đã chia sẻ trước giữa hai bên xác thực, hoặc các sự kiện diễn ra đồng thời ở cả hai bên.

Mật khẩu OTP còn gọi là mật khẩu động với đặc điểm không lặp lại (mật khẩu dùng mỗi lần sẽ không giống nhau) và chỉ có giá trị một lần, trong khoảng thời gian nhất định.

Phương pháp sử dụng kết hợp Username/Password thông thường với mật khẩu OTP sẽ đảm bảo hơn trong quá trình xác thực người dùng, còn được gọi là phương pháp xác thực hai nhân tố. Trong đó gồm mật khẩu cố định của người dùng, và mật khẩu OTP được sinh ra từ hệ thống cung cấp dịch vụ như hệ thống giao dịch ngân hàng, chứng khoán, thương mại điện tử,...

**1.2  Mục đích và ý nghĩa của mật khẩu OTP**

Mục đích của mật khẩu OTP là làm cho mật khẩu (OTP là mật khẩu động) lúc nào cũng thay đổi, tránh việc các *hackers* hoặc ai đó dễ dàng đánh cắp tài khoản của người dùng và sử dụng vào mục đích bất hợp pháp. Yêu cầu đối với mật khẩu OTP là có thể chống lại được cách thức tấn công phát lại, có nghĩa là trong trường hợp nào đó mà ai có được thông tin về OTP trong một phiên làm việc thì cũng không thể sử dụng nó để đăng nhập trong phiên làm việc kế tiếp, không thể đoán được mật khẩu tiếp theo để sử dụng.

Mật khẩu OTP có ý nghĩa quan trọng trong xác thực, OTP khi được sử dụng kết hợp với phương pháp xác thực gồm Username/Password sẽ đảm bảo an toàn cao hơn cho người dùng khi đăng nhập và xác thực với hệ thống.

Ví dụ mỗi khách hàng khi thực hiện giao dịch ngân hàng trực tuyến đầu tiên sẽ đăng nhập bằng tài khoản gồm Username/Password, sau đó để xác thực đúng tài khoản khách hàng đang thực hiện giao dịch thì ngân hàng sẽ sử dụng kết hợp với mật khẩu OTP. Mật khẩu OTP có thể được sinh ra thông qua thiết bị như Token được cấp cho khách hàng, hay được sinh ra từ hệ thống và gửi qua tin nhắn SMS tới số điện thoại, qua thư điện tử mà khách hàng đã đăng ký trước đó. Khách hàng sẽ sử dụng mật khẩu OTP nhận được để xác thực với ngân hàng và tiếp tục thực hiện giao dịch.

Với mật khẩu OTP có giá trị trong khoảng thời gian ngắn (thường là 30 giây) nếu có một ai đó có được Username/Password tài khoản giao dịch của khách hàng cũng không thể đăng nhập được vì cần phải nhập thêm mật khẩu OTP, mật khẩu này chỉ có thể có được khi sử dụng thiết bị Token của ngân hàng cấp hoặc phải có được tài khoản thư điện tử, số điện thoại của khách hàng có mật khẩu OTP mới có thể hoàn thành được quá trình xác thực.

Như vậy, có thể thấy được ý nghĩa trong quá trình xác thực người dùng của mật khẩu OTP, đặc biệt trong các giao dịch trực tuyến của ngân hàng, giao dịch trên các *Website* thương mại điện tử, giao dịch chứng khoán,...

**1.3  Yêu cầu đối với mật khẩu OTP**

Mật khẩu một lần (OTP) thường được sử dụng như là nhân tố xác thực thứ hai, giúp tăng tính an toàn trong việc xác thực và mật khẩu OTP chỉ có giá trị một lần. Tuy nhiên, nếu kẻ tấn công có khả năng đoán được giá trị OTP ở lần xác thực tiếp theo thì rõ ràng là OTP đã đánh mất vai trò của nó. Do vậy, yêu cầu tất yếu đặt ra khi sử dụng OTP là giá trị của OTP không thể đoán trước được bởi bên tấn công.

Hiện nay, có hai cách tiếp cận để giải quyết bài toán này: thứ nhất là OTP phải phụ thuộc vào một yếu tố bí mật và một yếu tố thay đổi theo thời gian và thứ hai là OTP được sinh một cách ngẫu nhiên.

**1.4  Phân loại cơ chế sử dụng mật khẩu OTP**

Về tổng thể, có 3 nhóm cơ chế sử dụng mật khẩu OTP: OTP được sinh ở phía claimant, OTP được sinh bởi cả claimant và verifier, OTP được sinh bởi verifier.

**OTP được sinh bởi claimant**: Mỗi khi tham gia vào phiên xác thực, claimant sẽ sinh ra một OTP và gửi đến verifer; verifer sẽ căn cứ vào giá trị OTP này để xác thực claimant. Ví dụ điển hình cho trường hợp này là các lược đồ xác thực sử dụng mật khẩu một lần Lamport.

**OTP được sinh bởi cả claimant và verifier**: Trong trường hợp này, hai bên xác thực sẽ thống nhất một lược đồ sinh OTP phụ thuộc vào một giá trị bí mật được chia sẻ giữa hai bên. Mỗi lần tham gia vào phiên xác thực, claimant sinh ra một OTP và gửi cho verifier. Bản thân verifier cũng sinh ra một OTP và so sánh với OTP nhận được từ claimant. Kết quả so sánh sẽ cho phép verifier kết luận về tính hợp lệ của claimant. Ví dụ cho cơ chế sử dụng OTP này là các lược đồ xác thực sử dụng OTP được quy định trong RFC-4226 (HOTP:HMAC- Based One-Time Password Algorithm).

**OTP được sinh bởi verifier**: Trong trường hợp này OTP được sinh bởi verifier và được gửi cho claimant qua kênh liên lạc thứ hai, khác với kênh được dùng để khởi xướng phiên xác thực. Kênh thứ hai này thường là kênh tin nhắn SMS, kênh thoại hoặc kênh email. Sau khi nhận được OTP từ verifier, claimant sẽ gửi trả lại cho verifier qua kênh thứ nhất. Verifier so sánh OTP nhận được từ claimant với OTP mà nó đã gửi đi. Nếu kết quả là giống nhau thì claimant được xác thực, ngược lại thì claimant bị từ chối. Có thể thấy rằng, về thực chất OTP trong trường hợp này được sử dụng để chứng minh sự sở hữu kênh liên lạc thứ hai của claimant (ví dụ: số điện thoại di động, hộp thư điện tử) mà đã được claimant đăng ký với verifier khi thiết lập tài khoản.

## 2. Xác thực với OTP được sinh ở phía claimant

Như đã trình bày trong mục 2.14 đại diện tiêu biểu cho cơ chế xác thực với OTP được sinh ở phía claimant là các lược đồ xác thực sử dụng mật khẩu một lần Lamport. Thuật toán Leslie Lamport được sử dụng để tạo ra mật khẩu mới dựa trên mật khẩu trước đó bằng cách áp dụng một hàm một chiều *f*. Hệ thống OTP làm việc dựa trên một giá trị mầm khởi tạo s để sinh mật khẩu lần đầu tiên, sau đó sinh ra các mật khẩu với các giá trị f(*s*), f(f(*s*)), f (f(f(*s*)))…fn(*s*), với *n* là số lần áp dụng hàm *f* lên giá trị mầm *s*. Ở đây cần lưu ý rằng các mật khẩu này khi đưa vào sử dụng thì được lấy theo thứ tự ngược lại vì vậy khi một người nào đó có mật khẩu sử dụng một lần chỉ sử dụng cho một phiên liên lạc duy nhất, nên không thể dùng mật khẩu này cho phiên liên lạc khác để truy cập vào hệ thống. Để có được mật khẩu trong dãy từ mật khẩu của lần truy cập trước đó để dùng cho phiên liên lạc tiếp theo, kẻ tấn công phải tim cách tính được giá trị hàm hàm nghịch đảo f-1. Vì hàm f là một hàm một chiều nên thực hiện được điều này là rất khó. Nếu hàm f-1 là một hàm băm mật mã a(cryptographic hash function) thì điều này gần như không thể thực hiện được. Như vậy, mật khẩu OTP được sử dụng cho phiên làm việc đầu tiên sẽ được tính như sau:

**OTP1 = fn(s)**                           (1)

Trong công thức (1): fn(s) = f(fn-1(s)), với *n* là số lần áp dụng hàm *f* lên giá trị mầm *s*.

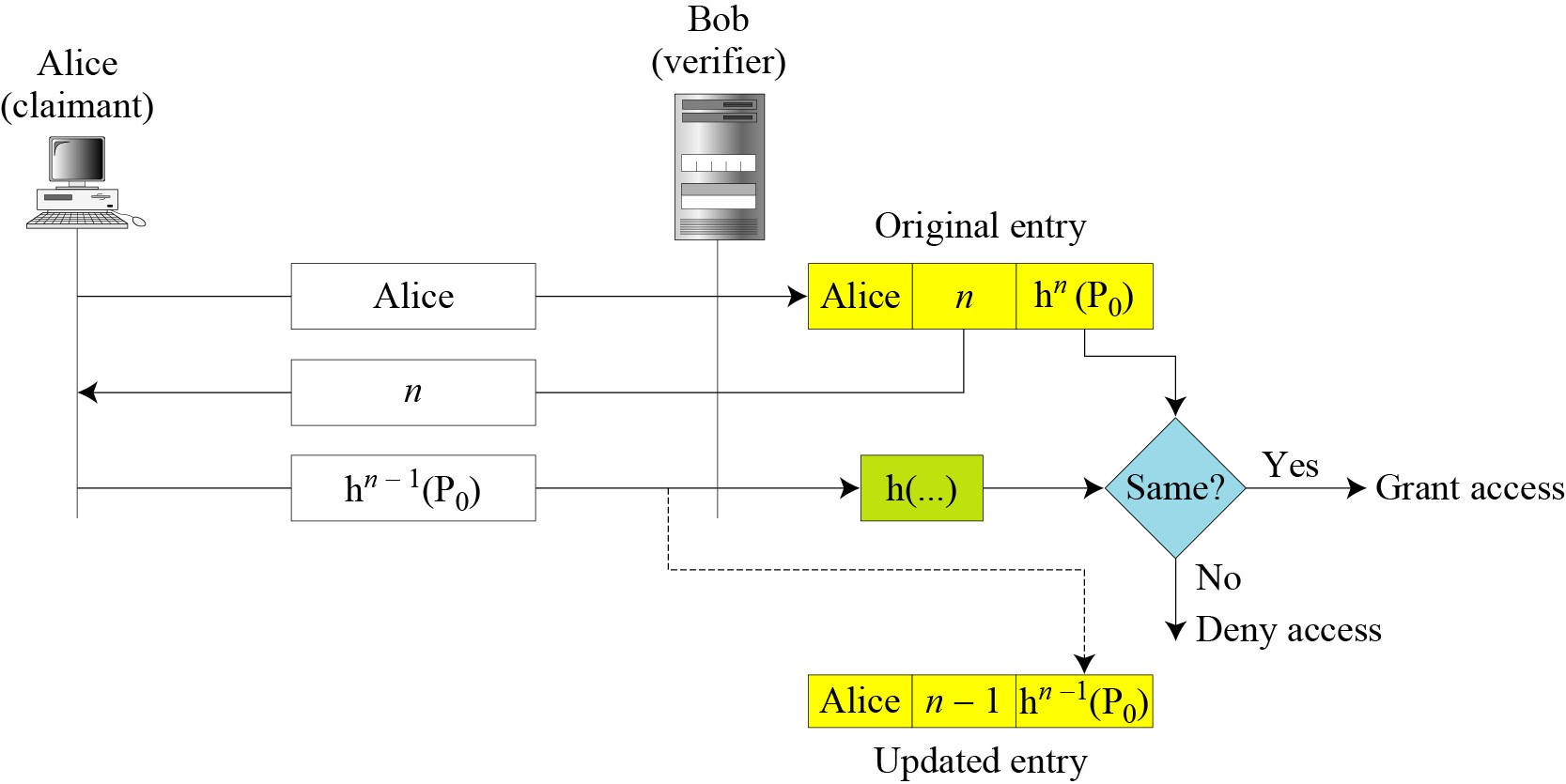
Theo như công thức (1) ta sẽ có lần lượt các mật khẩu OTP cho các phiên như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phiên 1 | Phiên 2 | Phiên 3 | … | Phiên n |
| OTP1=Fn(s) | OTP2=Fn-1(s) | OTP3=Fn-2(s) | … | OTPN=F1(s) |

*Bảng 1: Mật khẩu OTP cho các phiên sinh theo Lamport*

Việc ứng dụng mật khẩu một lần theo Lamport vào xác thực có thể được mô tả như theo hình 2.1. Theo đó, bên yêu cầu xác thực (claimant) là Alice và bên xác thực (verifier) là Bob, tức là ở đây ta có xác thực một chiều. Điều kiện ban đầu để xác thực gồm:

* Alice và Bob thỏa thuận mật khẩu bí mật của Alice là P0;
* Alice và Bob thỏa thuận sử dụng hàm băm *h*;
* Alice và Bob xác định (có thể căn cứ vào chính sách an toàn) số nguyên *n*0, đó là số lần xác thực mà Alice có thể thực hiện với mật khẩu P0; Alice ghi nhớ mật khẩu P0 của mình;
* Bob lưu giữ thông tin về Alice gồm: định danh (Alice), số *n n*0 , giá trị băm *n* lần của mật khẩu P0 là *Hn* = *h n* *P*0  .

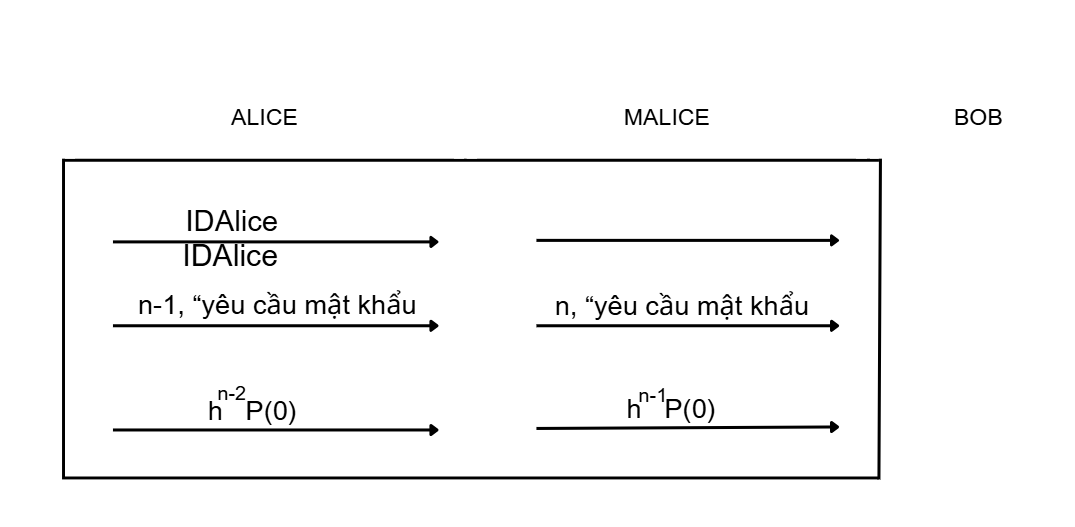


*Hình 2.1. Xác thực bằng mật khẩu một lần theo Lamport*

Với điều kiện ban đầu như trên, quá trình xác thực diễn ra như sau:

* Alice yêu cầu xác thực.
* Bob kiểm tra sự tồn tại định danh Alice. Nếu tồn tại Alice thì Bob gửi Alice số *n* tương ứng và chuyển đến Bước 3. Nếu Alice không tồn tại thì thông báo xác thực bất thành và chuyển đến Bước 5
* Alice băm mật khẩu P0 *n*–1 lần bằng hàm băm *h*, nhận được kết quả là *Hn*–1 = *hn*–1(P0) và gửi *Hn*–1 cho Bob
* Bob tiến hành kiểm tra *Hn*–1
* Bob dùng hàm băm *h* để băm giá trị *Hn*–1, thu được *h*(*Hn*–1)
* Bob so sánh *h*(*Hn*–1) với giá trị *Hn* ứng với định danh Alice trong CSDL. Nếu khác nhau thì thông báo xác thực bất thành và chuyển đến Bước 5.
* Thay giá trị *n* ứng với Alice bằng *n* – 1 Thay giá trị *Hn* ứng với Alice bằng *Hn*–1 Thông báo xác thực thành công.
* Kết thúc xác thực.

Có thể thấy rằng claimant sử dụng các mật khẩu (Hi) khác nhau trong các lần liên lạc khác nhau, mật khẩu của phiên liên lạc trước được verifier sử dụng để kiểm tra mật khẩu của phiên liên lạc trước. Kẻ tấn công nghe lén trên đường truyền có thể bắt được Hi nhưng không thể tìm ra được Hi-1 để giả mạo claimant trong phiên liên lạc kế tiếp là bởi vì *h* là một hàm băm mật mã, có tính một chiều. Tuy vậy, giao thức liên lạc được mô tả trên Hình 1 lại có thể bị tấn công bằng phương pháp MITD (man-in-the-middle) như theo hình 2.2

**

*Hình 2.2. Tấn công MITD lên giao thức xác thực sử dụng OTP theo Lamport*

Trong trường hợp được mô tả trên *Hình 2.2*, kẻ tấn công (Malice) thay đổi giá trị đếm (*n*) mà Bob gửi cho Alice, cụ thể là giảm đi một đơn vị, sau đó lại thay đổi giá trị OTP mà Alice gửi cho Bob, cụ thể là băm thêm một lần bằng hàm băm *h*, khiến cho Alice vẫn xác thực thành công với Bob. Tuy nhiên, sau phiên liên lạc này, Malice có thể sử dụng giá trị OTP chặn bắt được trước đó để giả mạo Alice để liên lạc với Bob.

Có thể ngăn chặn tấn công MITD.Tuy nhiên, việc này là rất khó khăn, gây ra sự bất tiện trong áp dụng thực tế.

## 3. Xác thực với OTP được sinh ở cả hai phía

**3.1  Sinh OTP dựa trên việc đồng bộ bộ đếm**

Thuật toán sinh OTP dựa trên đồng bộ bộ đếm được mô tả trong tài liệu RFC-4226 (HOTP: HMAC-Based One-Time Password Algorithm). HOTP là một thuật toán sinh mật khẩu OTP dựa trên hàm hash SHA-1. Đây là thuật toán đang được sử dụng nhiều, trở thành thuật toán tiêu chuẩn mở miễn phí.

Về cách sinh mật khẩu OTP sử dụng thuật toán HOTP: **HOTP(K,C) = Truncate(HMAC\_SHA-1(K,C))**        (2)

Trong công thức (2):

K: Là khóa bí mật, là giá trị chia sẻ bí mật giữa Client và Server.

C: Là bộ đếm đã được đồng bộ giữa Client và Server, C có độ dài 8 bytes.

Truncate(…): Là hàm tách chuỗi, thực hiện việc trích xuất kết quả từ hàm Hash để có được mật khẩu OTP (thường là 6 hoặc 8 chữ số). Phương pháp tách chuỗi như sau:

**HMAC\_SHA-1(K,C) = SHA-1(K⊕C1… ∥ SHA-1(K⊕ C2… ∥ C))** (3)

Trong công thức (3):

* K : Khóa bí mật.
* C : Bộ đếm.
* C1 : Hằng giá trị 0x36 (36 trong hệ thập lục phân).
* C2 : Hằng giá trị 0x5C (5C trong hệ thập lục phân).
* ⊕: Toán tử XOR (Exclusive OR) thực thi từng bit.
* ∥: Chuỗi kết nối toán tử (Concatenation)

HMAC\_SHA-1(K,C) là một hàm tính toán dựa trên thuật toán HMAC kết hợp với hàm băm SHA-1 của giá trị K và bộ đếm C.

Như chúng ta đã biết kết quả đầu ra của hàm HMAC\_SHA-1(K,C) cho ta một giá trị có độ dài là 160 bits = 20 bytes, chúng ta dùng hàm tách chuỗi (Truncate) để tách từ chuổi 160 bits thành một chuỗi mới có độ dài 32 bit, sau đó tính modul để được mật khẩu OTP. Cụ thể như sau:

Từ kết quả đầu ra 160 bit của hàm HMAC\_SHA1(K,C), ta lấy 4 bit thấp của byte cuối cùng chuyển sang cơ số 10 để tim vị trí offset, sau đó ta được chuỗi 4 bytes = 32 bit tính từ vị trí ofset.

Giá trị mật khẩu được tính theo công thức sau:

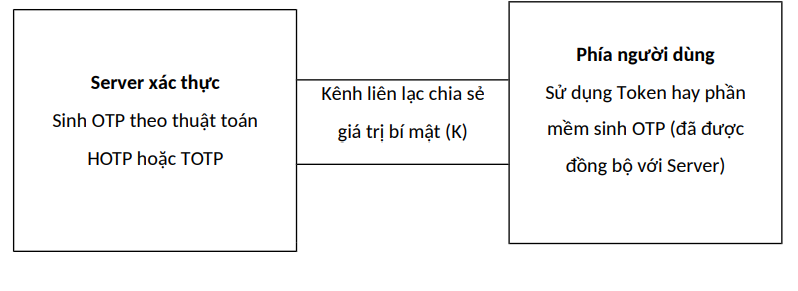
**HOTPvalue = HOTP(K,C) mod 10d** (4)

Trong công thức 4: *d* là số chữ số của OTP, thông thường một mật khẩu OTP sinh ra có độ dài từ 6 đến 8 chữ số (ví dụ như hình dưới là: 965083,...).



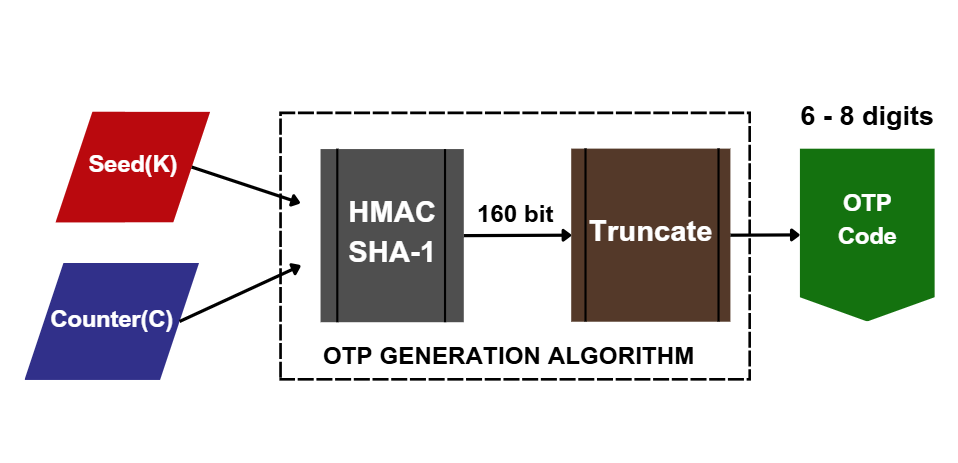
*Hình 2.3. Thiết bị phần cứng Token sinh mật khẩu OTP*

Ở đây, giá trị bí mật K được chia sẻ giữa máy chủ xác thực và máy trạm phía người dùng qua một kênh liên lạc nào đó. Sau đó máy chủ xác thực sử dụng giá trị bí mật K, bộ đếm C để sinh mật khẩu OTP theo công thức (2), máy trạm phía người dùng cũng sử dụng thiết bị sinh mật khẩu OTP như Token để sinh mật khẩu OTP (Token lưu giá trị bí mật K được chia sẻ và bộ đếm C). Để xác thực, máy chủ sẽ so sánh mật khẩu OTP sinh ra trên máy chủ với mật khẩu OTP sinh ra từ máy trạm phía người dùng, nếu giống nhau, người dùng trên máy trạm xác thực thành công.

**

*Hình 2.4. Chia sẻ giá trị bí mật khi sinh OTP ở cả hai bên xác thực*

Để hiểu được thuật toán sinh OTP dựa trên sự kết hợp của HMAC và hàm băm SHA-1 có thể phân tích quá trình sinh giá trị mật khẩu OTP được mô tả như sơ đồ đưới đây.



*Hình 2.5. Sơ đồ mô tả thuật toán HOTP sinh mật khẩu OTP*

Trong nhiều trường hợp, vấn đề xác thực giữa máy chủ xác thực và máy trạm phía người dùng sẽ sảy ra quá trình mất đồng bộ giá trị mật khẩu vì vậy, muốn người dùng đăng nhập hệ thống thực thành công, chúng ta có thể phân tích quá trình đồng bộ giá trị sinh mật khẩu OTP giữa máy chủ xác thực và máy trạm phía người dùng đối với thuật toán sinh OTP dựa trên sự kết hợp của HMAC và hàm băm SHA-1 như sau:

Thông thường, trên máy chủ xác thực của bên cung cấp dịch vụ cho người dùng sẽ thực hiện sinh mật khẩu OTP.

Trên máy trạm phía người dùng sẽ sử dụng ứng dụng sinh mật khẩu OTP, sau đó khi xác thực, máy chủ sẽ kiểm tra mật khẩu OTP sinh ra trên máy trạm từ phía người dùng có giống với mật khẩu OTP được sinh ra trên máy chủ hay không. Nếu mật khẩu OTP giống nhau, người dùng sẽ được xác thực và sử dụng dịch vụ của hệ thống. Trong một số trường hợp, vấn đề mất đồng bộ sảy ra khi người dùng sử dụng thiết bị sinh OTP như Token và ấn nút sinh OTP nhiều lần nhưng không sử dụng hoặc trong quá trình đăng nhập bị lỗi,… dẫn đến mật khẩu OTP sinh ra từ phía người dùng không giống với mật khẩu sinh ra từ phía máy chủ xác thực, khi đó người dùng xác thực không thành công.

Để giải quyết được vấn đề này, quá trình đồng bộ giá trị sinh mật khẩu OTP trên máy chủ xác thực và máy trạm phía người dùng sẽ được thực hiện như sau:

* Giả sử người dùng có một thiết bị phần cứng Token sinh mật khẩu OTP và mật khẩu OTP được bởi hàm: HMAC\_SHA1(Key, Counter\_Client).
* Máy chủ xác thực sinh mật khẩu OTP theo công thức: HMAC\_SHA1(Key,Counter\_Server).
* Người dùng muốn được xác thực và sử dụng dịch vụ của hệ thống thì phải đồng bộ Counter của máy trạm và máy chủ.
* Giả sử tại thời điểm hiện tại: Counter\_Client = Counter\_Server + a, với a<W (W là cửa sổ đồng bộ).
* Ban đầu, Client gửi OTP hiện thời tới Server.
* Server kiểm tra OTP vừa được gửi từ Client có là một giá trị trong tập các giá trị:

OTP=HMAC\_SHA1(Key,Counter\_Server+1)

OTP=HMAC\_SHA1(Key,Counter\_Server+2),...

OTP=HMAC\_SHA1(Key,Counter\_Server+W)

Trong đó sẽ có một giá trị OTP = HMAC\_SHA1 (Key, Counter\_Server + a) giống với của OTP từ máy trạm đã gửi lên. Máy chủ xác thực đúng máy trạm, sau đó cập nhật giá trị cho Counter lúc này là Counter\_Server = Counter\_Server + a. Tại thời điểm này máy trạm và máy chủ có cùng giá trị Counter, Counter được đồng bộ để cho các phiên xác thực sau.

**3.2  Sinh OTP dựa trên việc đồng bộ thời gian**

Đối với thuật toán HOTP đã trình bày ở phần trên, mật khẩu OTP được sinh ra dựa vào giá trị bí mật (K) được chia sẻ và một giá trị đếm (C). Còn đối với thuật toán TOTP được định nghĩa RFC 6238 là một thuật toán cũng tương tự như HOTP nhưng dựa trên thời gian T(Time) thay vì giá trị đếm C(Counter).

Với thuật toán TOTP sinh mật khẩu OTP dựa theo thời gian thì giá trị T về thời gian được tính như sau:

**T= [** (5)

Trong công thức (5):

* Tcurrent\_unix\_time: Là giá trị thời gian hiện tại được tính theo thời gian Unix time (được tính từ thời điểm của Unix Epoch là ngày 01/01/1970 theo UTC).
* T0: Là giá trị thời gian ban đầu (thường chọn T0 = 0).
* X: Là bước thời gian, đây là tham số quyết định thời gian hợp lệ của mật khẩu OTP (thường là 30 giây).
* T chính là kết quả tính (đã lấy phần nguyên) từ công thức tính toán trên.

**Ví dụ**

* **T0 = ​​0** (mốc thời gian bắt đầu là 0).
* **X = 30 giây** (OTP thay đổi sau mỗi 30 giây).
* **Tcurrent\_unix\_time = 1627598000** (thời gian hiện tại là 1627598000 giây tính từ Epoch).

Tính giá trị **T** :

T=[]=[ 52586599 ]=52586599

Sau đó, sử dụng giá trị **T** để tính HMAC-SHA1 và rút ra OTP.

Với thuật toán TOTP thì thời gian chuẩn hợp lệ của mật khẩu OTP là 30 giây (X=30), thời gian có hiệu lực của mỗi lần sử dụng mật khẩu OTP là 30 giây được chọn phù hợp với yêu cầu về bảo mật và khả năng sử dụng.

Thuật toán TOTP dựa trên thuật toán HOTP thay giá trị đếm (C) bằng giá trị thời gian (T):

**TOTP = HOTP(K,T)   (6)**

Đối với thuật toán TOTP độ dài của mật khẩu OTP được tính như sau:

**value = TOTP(K,T) mod   (7)**

Trong công thức (7):

* d là số chữ số của mật khẩu OTP, thông thường một mật khẩu OTP sinh ra có độ dài từ 6 đến 8 chữ số (ví dụ như : 968357,...).

Tương tự như vấn đề đã trình bày trong phần thuật toán HOTP, vấn đề mất đồng bộ về thời gian hay giá trị đếm khi sinh mật khẩu OTP có thể xảy ra dẫn đến mật khẩu sinh OTP trên máy chủ xác thực và máy trạm phía người dùng không giống nhau, người dùng sẽ xác thực không thành công để tiếp tục sử dụng dịch vụ. Khi đó, vấn đề đồng bộ lại thời gian hay giá trị đếm được thực hiện, đối với thuật toán TOTP khi sinh mật khẩu OTP là vấn đề đồng bộ về thời gian giữa máy trạm phía người dùng và máy chủ xác thực. Theo công thức (5) tính toán giá trị thời gian T cần phải có giá trị thời gian hiện tại Tcurrent\_unix\_time, nếu thời gian hiện tại không được đồng bộ thì người dùng trên máy trạm sẽ không xác thực hợp lệ được với máy chủ xác thực.

* Giả sử thời gian hiện tại trong đồng hồ bên trong thiết bị Token sinh mật khẩu OTP của máy trạm khi thực hiện việc xác thực với máy chủ xác thực là T1, và thời gian phía máy chủ là T'1. Như vậy, thời gian trên máy chủ với máy trạm không trùng khớp và chênh lệch.
* Giả sử thời gian ban đầu được chọn là T0 = 0 và bước thời gian X=30 giây, khi đó giá trị T trên máy trạm được tính là: TClient = [T1/30]
* Client sẽ sinh mật khẩu OTP dựa theo công thức HOTP(K,TClient).
* Đối với máy chủ xác thực thì: TServer = [T1'/30] và Server sẽ sinh mật khẩu một OTP dựa theo công thức HOTP(K,TServer).

Khi đó, mật khẩu một lần trên máy trạm dùng để xác thực với máy chủ sẽ không phù hợp và xác thực sẽ không thành công.

* Để việc xác thực được thành công, máy chủ sẽ thực hiện tính toán và thay đổi giá trị sao cho phù hợp.
* Client sinh mật khẩu OTP theo công thức: HOTP(K,TClient) không phù hợp với mật khẩu OTP sinh ra trên máy chủ: HOTP(K,TServer). Máy chủ sẽ tính giá trị khác lần lượt là: HOTP(K,TServer + *a*) với *a* là giá trị thời gian tính bằng giây.
* Khi tính đến một giá trị *a* nào đó như *a*=*a*1 và OTP= HOTP(K,TServer + a1) thấy giống với mật khẩu OTP được sinh ra trên máy trạm, máy chủ sẽ cập nhật lại thời gian và cho phép người dùng trên máy trạm xác thực. Máy chủ có thể tính toán thời gian T và điều chỉnh cho phù hợp theo đồng hồ riêng.

***Ví dụ về sinh mật khẩu OTP theo thuật toán HOTP****:*

Giả sử tính được HMAC\_SHA-1(Secret(K), Time(T)) = 1f | 86 | 98 | 69 | 0e | 02 | ca | 16 | 61 | 85 | 50 | ef | 7f | 19 | da | 8e | 94 | 5b | 55 | 5a

Giá trị được tạo ra sau khi sử dụng HMAC-SHA-1 có độ dài 160 bits chính là mã xác thực thông điệp MAC.

Sau đó, sử dụng hàm tách chuỗi Truncate(HMAC\_SHA-1(K,C)) mã này ta sẽ có một chuỗi mới có độ dài 32 bit. Cụ thể tách như sau:

* Lấy 4 bits thấp của byte cuối cùng, trong ví dụ trên thì byte cuối cùng là một giá trị hexa là 5a (01011010 theo cơ số 2) 4 bits thấp nhất của byte cuối cùng này là 1010 tương ứng là 10 (trong hệ cơ số 10).
* Lấy tiếp 4 bytes từ vị trí Ofset 10 (đếm từ 0) ta sẽ có: 1f |86 | 98 | 69 | 0e | 02 | ca | 16 | 61 | 85 | **50 | ef | 7f | 19** | da | 8e | 94 | 5b |55 | 5a.
* Giá trị có được là: 0x50ef7f19, đổi sang cơ số 10 ta được giá trị mới là: 1357872921. Đây chính là chuỗi vừa được hàm truncate() tách ra từ mã xác thực thông điệp MAC.

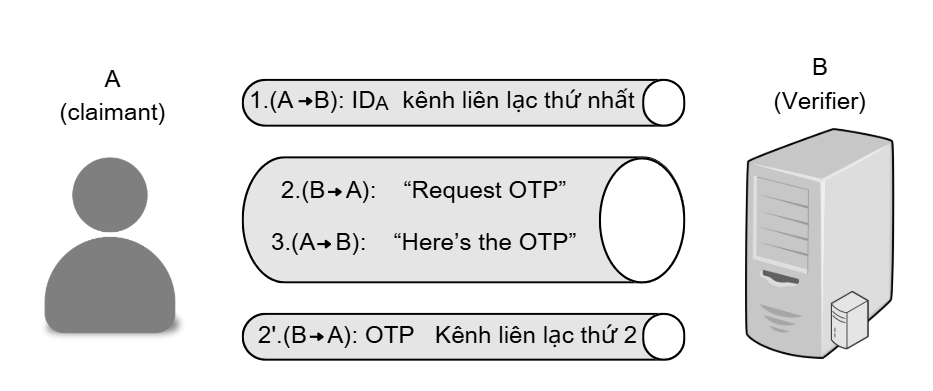
Thông thường mật khẩu OTP hiện nay có độ dài từ 6 đến 8 chữ số, theo thuật toán sinh OTP dựa trên đồng bộ về thời gian (TOTP) ta có thể có được mật khẩu OTP mới có độ dài tùy chọn theo công thức (7)

Dựa vào công thức (7) sinh được mật khẩu OTP có độ dài 6 chữ số tương ứng là: **872921**.Đây chính là mật khẩu OTP được sinh ra và hiển thị trên ứng dụng của người dùng, mật khẩu OTP này sẽ thay đổi theo thời gian (mặc định là sau 30 giây sẽ thay đổi). Người dùng sẽ nhập mật khẩu này để xác thực với máy chủ xác thực. Máy chủ xác thức sẽ sinh mật khẩu OTP và kiểm tra mật khẩu OTP được sinh ra trên ứng dụng phía người dùng, nếu giống nhau người dùng xác thực thành công. Trong một số trường hợp, vấn đề mất đồng bộ xảy ra và việc đồng bộ lại đã được trình bày ở trên là thực hiện đồng bộ lại về thời gian hoặc bộ đếm.

## 4  Xác thực với OTP được sinh ở phía verifier

**4.1  Sơ đồ tổng thể của xác thực với OTP được sinh ở phía verifier**

Có thể thấy rằng các phương pháp sinh OTP trên đây đều đòi hỏi cả hai bên tham gia vào việc xác thực (cả bên yêu cầu xác thực và bên xác thực) cùng phải có khả năng sinh OTP. Trong một số trường hợp, phía yêu cầu xác thực không có khả năng (vì lý do kinh tế, kỹ thuật) để sinh OTP. Khi đó, giải pháp khác được sử dụng là bên xác thực sẽ sinh OTP và gửi cho bên yêu cầu xác thực qua một kênh liên lạc thứ hai. Giá trị OTP trong trường hợp này phải được sinh một cách ngẫu nhiên. Kênh liên lạc thứ hai được coi là an toàn, tức là OTP sẽ không bị chặn bắt bởi kẻ tấn công đang hoạt động trên kênh liên lạc chính.



*Hình 2.6. Sơ đồ tổng thể của xác thực với OTP được sinh ở phía verifer*

Sơ đồ tổng thể của xác thực với OTP được sinh ở phía verifier có thể được mô tả như trên Hình 2.6. Trong đó, kênh liên lạc thứ hai chỉ được sử dụng để truyền OTP từ verifier đến claimant. Quá trình xác thực diễn ra như sau:

* A khởi xướng phiên xác thực qua kênh liên lạc thứ nhất, yêu cầu được giao tiếp với B với tài khoản có định danh là IDA (B sử dụng thông tin này để xác định A là ai trong hệ thống).
* Qua kênh liên lạc thứ hai, B gửi cho A một mã OTP; đồng thời, qua kênh liên lạc thứ nhất, B yêu cầu A gửi lại cho B mã OTP kể trên.
* Qua kênh liên lạc thứ nhất, A gửi cho B mã OTP nhận được qua kênh thứ hai để chứng minh tính sở hữu của A đối với kênh này.
* B so sánh OTP nhận được từ A (qua kênh liên lạc thứ nhất) với OTP đã gửi và đưa ra kết quả xác thực.

Có thể thấy rằng, nếu giả thiết kênh liên lạc thứ hai là an toàn (OTP không thể bị chặn bắt bởi bên thứ ba) thì vấn đề an toàn đối với OTP trong trường hợp này là cần đảm bảo rằng OTP không thể bị đoán trước, tức OTP phải là ngẫu nhiên. Bài toán sinh OTP như vậy có thể được quy về bài toán sinh số ngẫu nhiên.

Hiện nay có nhiều phương pháp sinh OTP. Trong nội dung luận văn tốt nghiệp, học viên tìm hiểu thuật toán sinh số ngẫu nhiên tất định (DRBG: Deterministic Random Bit Generator) hay còn g ọi là thuật toán sinh số giả ngẫu nhiên (PRNG: Pseudorandom Number Generator) được Viện tiêu chuẩn và công nghệ quốc gia Hoa Kỳ giới thiệu trong chuẩn SP 800-90A, tháng 1 năm 2012.

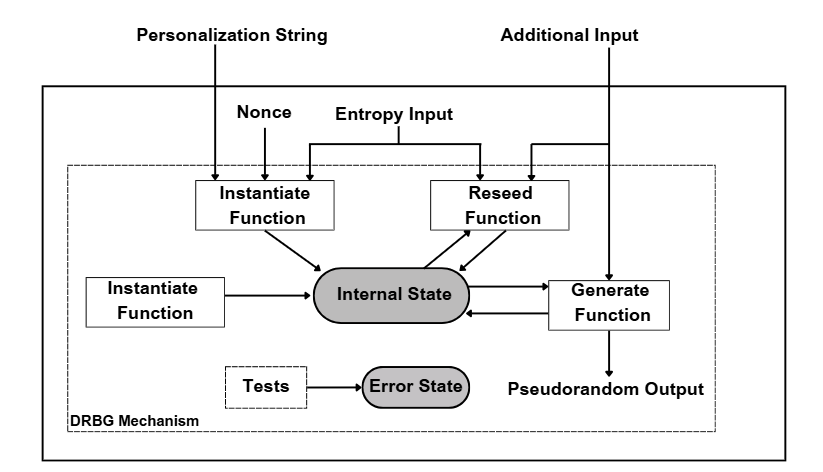
**4.2  Sinh số ngẫu nhiên theo NIST SP 800-90A**

Khuyến nghị này giới thiệu kỹ thuật để tạo ra các bits ngẫu nhiên sau đó có thể được sử dụng trực tiếp hoặc chuyển đổi sang số ngẫu nhiên khi các giá trị ngẫu nhiên được yêu cầu để sử dụng trong các ứng dụng mật mã.

Mô hình chức năng của một DRBG được thể hiện trên Hình 2.7. Mô hình này bao gồm một cơ chế DRBG (DRBG Mechanism) và một nguồn ngẫu nhiên thực sự (Entropy Input) đóng vai trò là đầu vào. Ngoài ra, tùy thuộc vào cài đặt cụ thể của DRBG Mechanism mà mô hình có thể bao gồm một nguồn nounce.

Nguồn ngẫu nhiên đầu vào (Entropy Input) cần phải được giữ bí mật, bởi nó cung cấp lượng entropy cần thiết cho DRBG Mechanism và do đó quyết định tính an toàn đối với dãy bít đầu ra.

Tùy theo cài đặt mà DRBG có thể có các đầu vào khác: Nounce, chuỗi cá nhân hóa (Personalization String) và đầu vào bổ sung (Additional Input). Các đầu vào này có thể được giữ bí mật hoặc không, tuy nhiên tính an toàn của DRBG không phụ thuộc vào tính bí mật của các đại lượng này. NIST khuyến cáo sử dụng Personalization String để đảm bảo DRBG sinh ra những chuỗi ngẫu nhiên khác nhau khi khởi chạy cho những đối tượng khác nhau.



*Hình 2.7. Mô hình chức năng của một DRBG*

Nội trạng thái (Internal State) là bộ nhớ của DRBG. Nó chứa tất cả các tham số, các biến và các giá trị khác được sử dụng bởi DRBG. Nội trạng thái này bao gồm cả các thông tin cấu hình (ví dụ: độ dài tối đa của chuỗi bít đầu ra) và các dữ liệu khác được thay đổi trong quá trình hoạt động.

DRBG Mechanism có 5 hàm chức năng khác nhau. Các hàm này đóng các vai trò khác nhau trong các thao tác lên nội trạng thái của DRBG.

* Hàm khởi tạo (Instantiate Function) lấy một giá trị ngẫu nhiên từ nguồn ngẫu nhiên đầu vào (entropy input) và có thể kết hợp với nounce và chuỗi cá nhân hóa (personalization string) để sinh ra một giá trị gieo mầm (seed). Giá trị seed này sẽ quyết định nội trạng thái (Internal State) ban đầu của DRBG.
* Hàm sinh (Generate Function): khi được yêu cầu, hàm sinh sẽ căn cứ vào nội trạng thái hiện tại để sinh ra các bít giả ngẫu nhiên, đồng thời thay đổi nội trạng thái của DRBG để phục vụ cho các yêu cầu sinh bít giả ngẫu nhiên kế tiếp.
* Hàm tái gieo mầm (Reseed Funtion): khi được gọi, hàm này sẽ sử dụng một giá trị ngẫu nhiên từ nguồn ngẫu nhiên đầu vào, kết hợp với nội trạng thái hiện thời và các đầu vào bổ sung khác nếu có để tạo ra một giá trị gieo mầm (seed) mới và một nội trạng thái mới cho DRBG.
* Hàm xóa trạng thái (Uninstantiate Function): khi được gọi, hàm này sẽ quy không (tức là xóa) nội trạng thái của DRBG.
* Hàm kiểm tra (Health Test Function) có chức năng kiểm tra tính đúng đắn của hoạt động của DRBG Mechanism.

Hoạt động các hàm trên đây, cũng như các cài đặt cụ thể của chúng được trình bày chi tiết trong NIST SP 800-90A.

Ở đây chúng ta sẽ làm rõ hoạt động của hàm sinh chuỗi bít giả ngẫu nhiên (Generate Function). Hàm này được gọi sau khi khởi tạo hoặc sau khi tái gieo mầm. Thuật toán sinh chuỗi bít có thể được mô tả như sau:

**Cú pháp hàm sinh chuỗi bits:**

**Generate\_function**(*state\_handle,requested\_number\_of\_bits,requested\_zecurity\_strength,prediction\_resistance\_request, additional\_input*)

**Mô tả đầu vào**

* *state\_handle*: Là một con trỏ hoặc một chỉ số để biểu thị nội trạng thái của DRBG.
* *requested\_number\_of\_bits*: Số lượng bít giả ngẫu nhiên sẽ được sinh ra. Giá trị này phải nhỏ hơn *max\_number\_of\_bits\_per\_request*. Bản thân giá trị tối đa này phụ thuộc vào cài đặt và phải nhỏ hơn chu kỳ của chuỗi giả ngẫu nhiên được sinh ra sau mỗi lần reseed.
* *requested\_security\_strength*: Mức an toàn được yêu cầu. Nếu DRBG chỉ hỗ trợ một mức an toàn thì có thể không sử dụng tham số này. Tuy nhiên, ứng dụng sử dụng DRBG cần phải nắm được mức an toàn mà DRBG hỗ trợ.
* *prediction\_resistance\_request*: cho biết có cần phải thực thi chức năng chống dự đoán hay không.
* *additional\_input*: đầu vào bổ sung mang tính tùy chọn.

**Mô tả Đầu ra:**

* *status:* Trạng thái của kết quả hoạt động của hàm có thể là success hoặc error*.*
* *pseudorandom\_bits*: Chuỗi bit giả ngẫu nhiên mà ứng dụng yêu cầu.

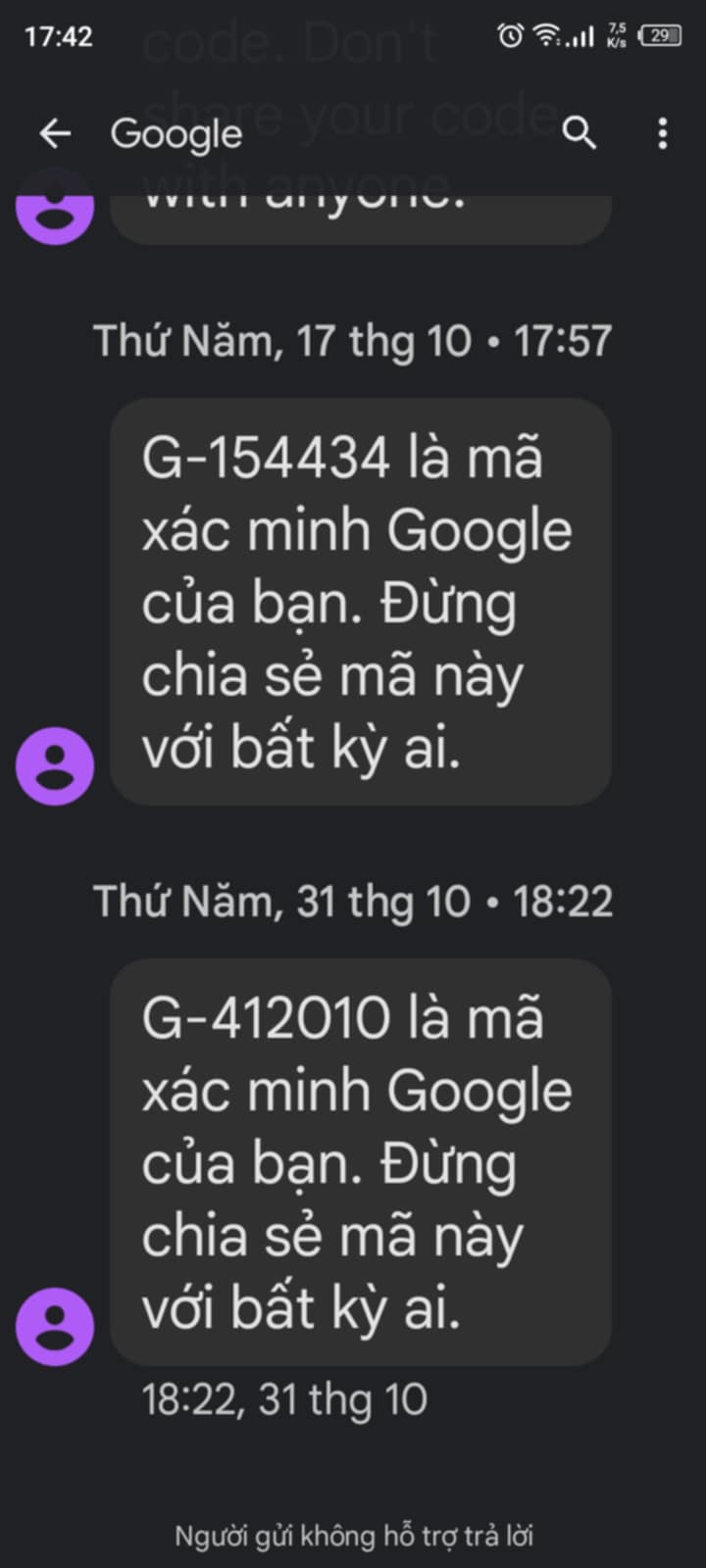
**Quá trình hoạt động:**

* Sử dụng tham số state\_handle để xác định nội trạng thái của DRBG. Nếu state\_handle trỏ đến một trạng thái không hợp lệ hoặc không khả dụng thì trả về ERROR\_FLAG.
* Nếu *requested\_number\_of\_bits* vượt giá trị *max\_number \_of\_bits\_per\_request*thì trả về ERROR\_FLAG.
* Nếu *requested\_security\_strength* lớn hơn giá trị *security\_strength* được xác định trong nội trạng thái của DRBG thì trả về ERROR\_FLAG.
* Nếu kích thước của *additional\_input* vượt giá trị *max\_additional\_input\_length.*  thì trả về ERROR\_FLAG.
* Nếu *prediction\_resistance\_request* được bật mà cờ *prediction\_resistance\_flag.* không được bật thì trả về ERROR\_FLAG.
* Nếu cờ *reseed\_required\_flag* được bật, hoặc cờ *prediction\_resistance\_request* được bật thì: *status=Reseed\_function(state\_handle,prediction\_resistance\_request, additional\_input)*. Nếu status là lỗi thì trả về ERROR\_FLAG
* Sử dụng *state\_handle* để biết nội trạng thái mới *additional\_input = null\_string*
* Xóa cờ *reseed\_required\_flag(status,pseudorandom\_bits, new\_working\_state)=****Generate\_algorithm****(working\_state,requested\_number\_of\_bits,additional\_ input)*
* Nếu status cho biết cần phải tái gieo mầm trước khi sinh các bít giả ngẫu nhiên thì: Bật cờ *reseed\_required\_flag*. Nếu cờ *prediction\_resistance\_flag* được bật thì bật cờ *prediction\_resistance\_request*. Thay giá trị *working\_state* cũ trong nội trạng thái của DRBG (được trỏ đến bởi *state\_handle*) bằng giá trị *new\_working\_state*.
* Trả về SUCCESS và *pseudorandom\_bits*.

Từ kết chuỗi bít giả ngẫu nhiên thu được từ thuật toán trên đây, ta có thể chuyển đổi thành mã OTP để sử dụng trong lược đồ xác thực với OTP được sinh ở phía verifier.

**4.3  Phương pháp phân phối OTP**

Phương pháp phân phối mật khẩu OTP tới người dùng phổ biến hiện nay là thông qua tin nhắn SMS tới số điện thoại đã đăng ký, thư điện tử tới địa chỉ Email người dùng đã đăng ký trước đó để thực hiện các giao dịch.

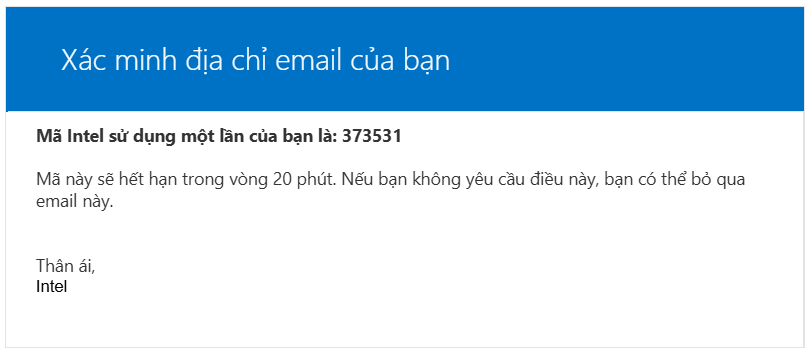


*Hình 2.8. Tin nhắn SMS cung cấp mật khẩu OTP*

Với phương pháp phân phối OTP qua tin nhắn SMS là phương pháp phân phối phổ biến hiện nay, được nhiều người dùng lựa chọn do tính tiện lợi của nó.

Các nhà mạng di động hiện nay cho phép khách hàng thực hiện giao dịch tiện lợi, nhanh chóng hơn qua các thiết bị di động. Khách hàng có thể nhận được mật khẩu OTP khi xác thực sử dụng dịch vụ, với mỗi lần xác thực sẽ có tin nhắn gửi mật khẩu sử dụng cho khách hàng. Những mật khẩu này cũng có thời gian hợp lệ nhất định, yêu cầu khách hàng nhập mật khẩu trong thời gian đó để hoàn thành giao dịch nếu không sẽ phải thực hiện lại giao dịch và có tin nhắn mới về mật khẩu OTP.

Với phương pháp thức phân phối OTP tới người dùng qua Email vẫn được sử dụng, tuy nhiên hiện nay gửi OTP qua Email ít hơn do tính tức thời của nó kém, ví dụ như khi khách hàng sử dụng điện thoại thông thường không có chức năng kiểm tra Email để giao dịch ngay, trong khi OTP chỉ có hợp lệ trong khoảng thời gian nhất.



*Hình 2.9. Email cung cấp mật khẩu OTP*

Hai phương pháp phân phối OTP tới người dùng qua tin nhắn và Email chủ yếu áp dụng trong trường hợp chỉ sinh mật khẩu OTP trên Server xác thực của nhà cung cấp dịch vụ.

Đối với các giao dịch cần có độ an toàn cao, việc sử dụng Token sinh OTP thường được khuyến khích sử dụng hàng nhằm nâng cao độ bảo mật thông tin cho khách hàng (Token sinh mật khẩu OTP sử dụng trong trường hợp cả phía Server xác thực và người dùng đều thực hiện sinh OTP và so sánh khi xác thực). Với phương thức nhận OTP qua tin nhắn SMS không thực sự đảm bảo an toàn vì thực tế đã xảy ra trường hợp kẻ tấn công nắm được thông tin về tài khoản của khách hàng, lợi dụng sơ hở từ phía nhà cung cấp di động nhằm chiếm đoạt SIM điện thoại của khách hàng và thực hiện đánh cắp tài khoản người dùng. Hơn nữa, với phương thức phân phối OTP qua SMS thì các tin nhắn không được mã hóa, khó tránh khỏi các nguy cơ mất an toàn khi vấn đề bảo mật đường truyền giữa nhà cung cấp dịch vụ giao dịch trực tuyến và nhà mạng di động không đảm bảo an toàn.

Có thể thấy rằng, hiện nay người dùng sử dụng Internet chủ yếu xác thực thông tin bằng hình thức xác thực định danh (Username/Password). Do đó, việc tăng cường hình thức bảo mật cho Password là vấn đề rất được quan tâm và mật khẩu OTP (One-Time- Password) là một giải pháp rất hữu hiệu. Trong nội dung luận văn đã trình bày ba cơ chế để sinh được mật khẩu OTP. Tuy nhiên, trong các cơ chế đó đều có những ưu và nhược điểm riêng.

Đối với cơ chế sinh mật khẩu OTP ở phía Claimant mặc dù đã sử dụng hàm Băm để mã hóa mật khẩu nhưng vẫn tồn tại khả năng bị tấn công theo phương pháp MITD (man- in-the-middle).

Đối với cơ chế sinh mật khẩu OTP ở phía Verifier thì mật khẩu OTP có tính bảo mật cao, khó bị các hacker tấn công. Tuy nhiên, để sử dụng mật khẩu OTP này đòi hỏi phải thiết lập một kênh liên lạc thứ hai (có thể là qua tin nhắn SMS, …) điều này đồng nghĩa với việc người dùng sẽ phải trả thêm cước phí để duy trì kênh liên lạc đó, gây ít nhiều bất tiện cho người dùng.

Đối với cơ chế sinh mật khẩu OTP ở cả hai phía thì độ an toàn của mật khẩu OTP là rất cao lại rất thuận lợi cho người dùng khi cài đặt và sử dụng do đó học viên đã lựa chọn nghiên cứu cơ chế sinh mật khẩu OTP ở cả hai phía theo thuật toán TOTP (Time - Based One Time Password) để áp dụng cho việc xác thực thông tin người dùng trong hệ thống học tập trực tuyến sẽ được trình bày ở Chương 3.

## 5  Một số ứng dụng OTP trong thực tế hiện nay

**5.1  Ứng dụng OTP trong giao dịch ngân hàng**

Thực tế hiện nay, mật khẩu OTP được sử dụng nhiều trong việc hỗ trợ xác thực khi người dùng đăng nhập để thực hiện các giao dịch của ngân hàng như rút tiền, chuyển khoản,...

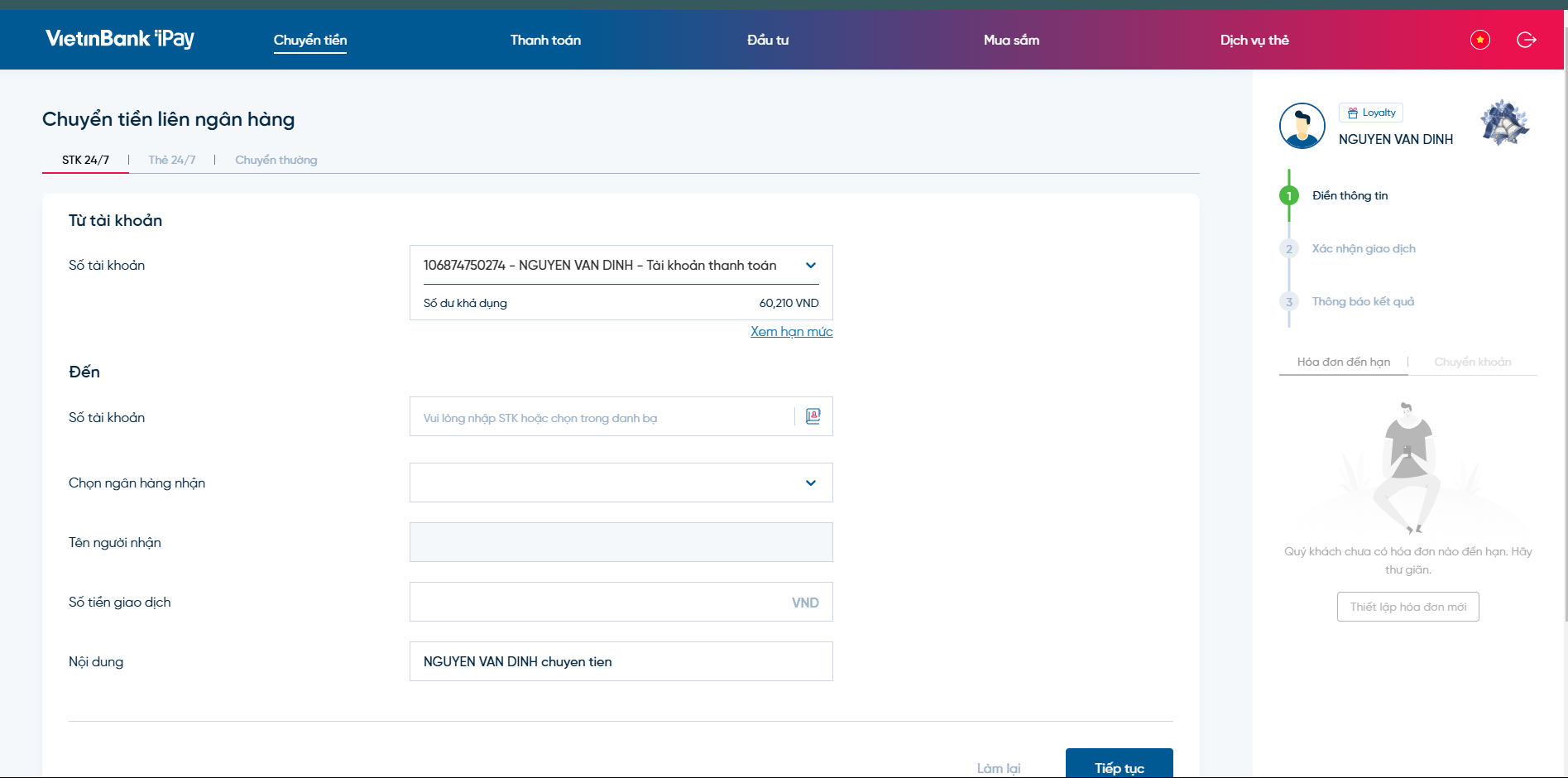
Mật khẩu OTP thường được sử dụng kết hợp với các phương pháp xác thực cơ bản sử dụng Username/Password, mật khẩu OTP được sinh ra và chỉ có hiệu lực một lần trong khoảng thời gian nhất định, nó sẽ tránh được rủi ro cho người dùng khi thực hiện giao dịch ngân hàng khi có người khác biết được mã PIN, mật khẩu tài khoản nhưng sẽ khó có thể đăng nhập vì cần phải có mật khẩu OTP.

Mật khẩu OTP được sử dụng một lần và có hiệu lực trong thời gian ngắn (thường vài chục giây). Người dùng hợp pháp nhận OTP qua các thiết bị như Token do ngân hàng cấp, SMS, hoặc email đã đăng ký. Nhờ đó, dù kẻ tấn công biết mã PIN hoặc mật khẩu, họ khó truy cập tài khoản nếu không có OTP. Chỉ khi kẻ tấn công có cả mã PIN, mật khẩu và thiết bị nhận OTP mới có thể xác thực thành công.

Hiện nay, các ngân hàng cung cấp dịch vụ Internet Banking, cho phép khách hàng thanh toán trực tuyến mà không cần đến ngân hàng. Để bảo vệ tài khoản khỏi các nguy cơ như giả mạo và lừa đảo, các ngân hàng áp dụng giải pháp xác thực bằng mật khẩu OTP, được gửi qua thiết bị Token, tin nhắn SMS, hoặc email.

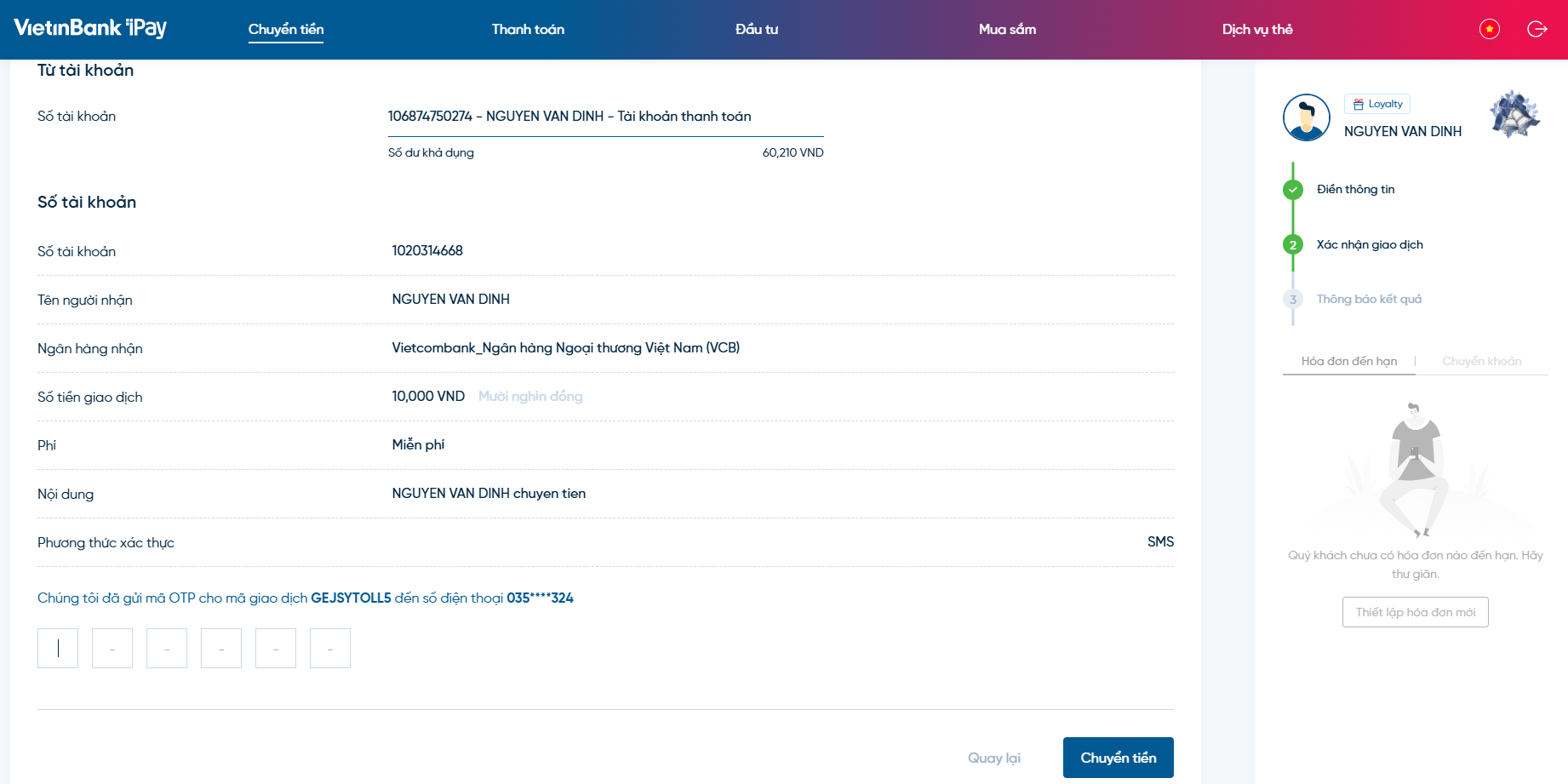
Dưới đây là một hướng dẫn cụ thể của ngân hàng để thực hiện một giao dịch có sử dụng mật khẩu OTP:

* Sử dụng Mobile OTP để xác thực các giao dịch trên Internet Banking của ngân hàng Eximbank
* Khách hàng lựa chọn thực hiện giao dịch trên Internet Banking của ngân hàng (ví dụ như lựa chọn dịch vụ chuyển khoản).



*Hình 2.10. Lựa chọn dịch vụ của ngân hàng để giao dịch*

* Khách hàng tiến hành nhập thông tin cần thiết và lựa chọn phương pháp xác thực sử dụng là Mobile OTP.
* Khách hàng nhập mật khẩu OTP đã được sinh ra trên ứng dụng Mobile OTP mà ngân hàng đã cung cấp để xác nhận giao dịch.

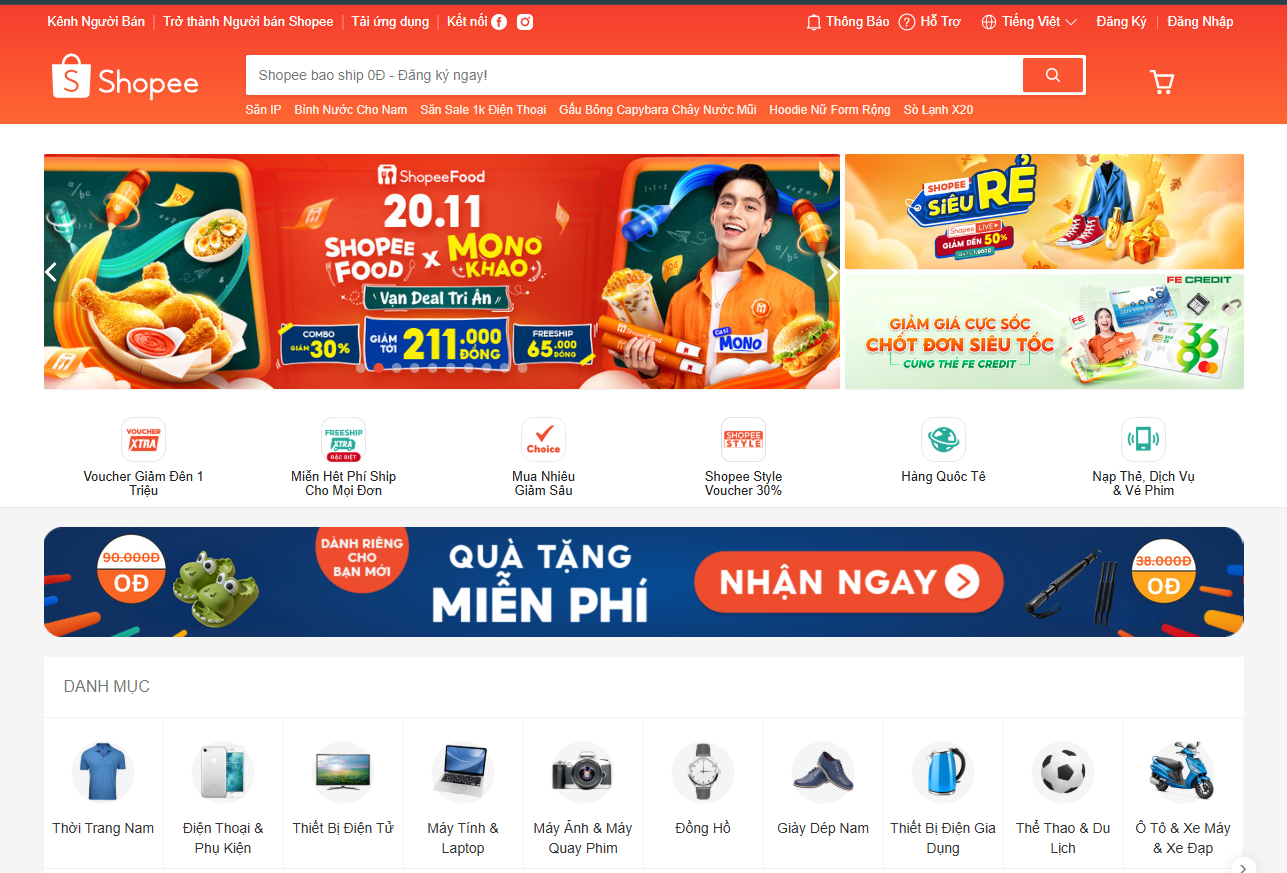
*Hình 2.11. Xác nhận giao dịch ngân hàng*

* Khách hàng nhập mã OTP và chọn "Chuyển tiền” để hoàn tất giao dịch.

Hiện nay, với dịch vụ sử dụng Internet Banking hay Mobile Banking một ngân hàng sẽ thực hiện gửi mật khẩu OTP về số điện thoại đã đăng ký của khách hàng, khách hàng sẽ sử dụng mật khẩu OTP này để xác minh trong các giao dịch của ngân hàng. Trong một số trường hợp khác, ngân hàng có ứng dụng riêng để sinh mật khẩu OTP trên điện thoại di động của khách hàng và sẽ được đồng bộ OTP với hệ thống ngân hàng. Mật khẩu OTP chỉ có giá trị hợp lệ trong một khoảng thời gian nhất định, nên nếu khách hàng không nhập ngay mật khẩu OTP trong khoảng thời gian đó thì giao dịch không được hoàn tất và sẽ quay trở lại thực hiện từ đầu.

**5.2. Ứng dụng (OTP) cho hệ thống giao dịch trực tuyến**

Mật khẩu OTP không chỉ được ứng dụng phổ biến trong giao dịch ngân hàng mà còn trong các lĩnh vực khác như thương mại điện tử, chứng khoán, v.v. Trong thương mại điện tử, các website được thiết kế để hỗ trợ các giao dịch mua bán trực tuyến, giúp khách hàng dễ dàng chọn lựa sản phẩm và thanh toán mà không cần đến trực tiếp cửa hàng. Hệ thống thanh toán trực tuyến trên các website thương mại điện tử cho phép người dùng thực hiện giao dịch thông qua các phương thức như thẻ tín dụng, thẻ ghi nợ, ví điện tử hoặc các cổng thanh toán được liên kết với ngân hàng. Việc mua sắm và thanh toán trên môi trường trực tuyến mang lại nhiều tiện lợi, nhưng cũng đi kèm với những rủi ro tiềm ẩn như lừa đảo, giả mạo, hay tấn công mạng để đánh cắp thông tin tài khoản cá nhân.

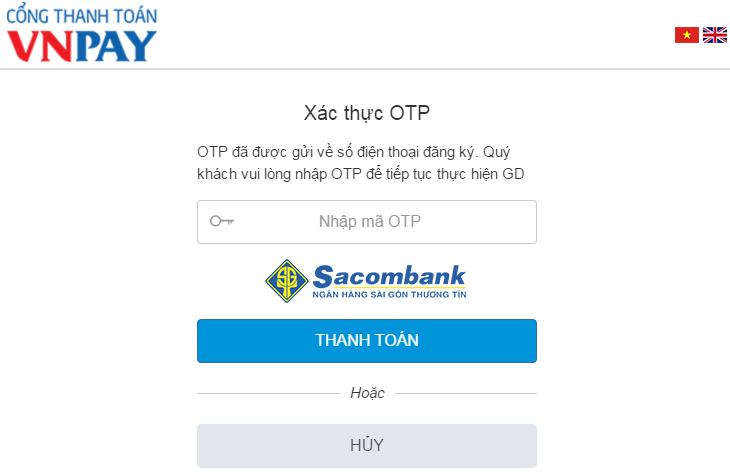


*Hình 2.12. Website Thương mại điện tử*

Các hình thức tấn công phổ biến trên nền tảng thương mại điện tử bao gồm phishing, trong đó kẻ xấu giả mạo email hoặc website để lừa người dùng cung cấp thông tin đăng nhập, tài khoản ngân hàng hoặc thông tin thẻ tín dụng. Ngoài ra, các tấn công sử dụng phần mềm keylogger để ghi lại thao tác bàn phím hoặc brute force để thử nhiều mật khẩu cũng được sử dụng nhằm xâm nhập tài khoản người dùng. Do đó, việc quản lý rủi ro và tăng cường bảo mật trong thanh toán trực tuyến trở thành một yêu cầu thiết yếu.

Trước đây, nhiều hệ thống thanh toán chỉ sử dụng xác thực đơn nhân tố dựa trên Username và Password. Tuy nhiên, phương pháp này đã dần trở nên lỗi thời khi các hành vi lừa đảo ngày càng tinh vi. Hiện nay, các hệ thống đã chuyển sang áp dụng xác thực đa nhân tố để tăng cường bảo mật, đặc biệt là trong các giao dịch thanh toán trực tuyến. Xác thực đa nhân tố thường kết hợp nhiều yếu tố như mật khẩu, số định danh cá nhân (PIN), chứng thư số, thiết bị bảo mật vật lý (Token sinh OTP), hoặc các yếu tố sinh trắc học (vân tay, nhận diện khuôn mặt). Phương pháp xác thực hai nhân tố, trong đó kết hợp Username/Password với OTP, là một giải pháp phổ biến nhờ tính hiệu quả và tiện lợi.

Trên các website thương mại điện tử, OTP thường được gửi đến người dùng qua SMS hoặc email đã đăng ký, hoặc thông qua thiết bị phần cứng Token. Cách này không chỉ đảm bảo rằng chỉ người dùng hợp pháp mới nhận được mã OTP, mà còn giúp bảo vệ tài khoản khách hàng ngay cả khi Username/Password bị lộ. Sử dụng xác thực hai nhân tố không chỉ an toàn mà còn thuận tiện, phù hợp với các dịch vụ như e-Banking, Internet Banking, hay Mobile Banking.



*Hình 2.13. Cổng thanh toán sử dụng xác thực với OTP*

Ngoài ra, các cổng thanh toán trực tuyến uy tín như PayPal, OnePay cũng tích hợp cơ chế xác thực đa nhân tố, kết hợp mã OTP với giao thức mã hóa SSL để bảo vệ thông tin giao dịch. Đây là một trong những biện pháp bảo mật phổ biến nhất, giúp ngăn chặn các rủi ro từ Internet và đảm bảo các giao dịch thương mại điện tử diễn ra an toàn.

Thông thường, quá trình sử dụng mật khẩu một lần OTP khi khách hàng thực hiện một thanh toán trực tuyến sẽ gồm một số bước như sau:

* Bước 1: Khách hàng lựa chọn mặt hàng cần mua sau đó thực hiện thanh toán.
* Bước 2: Chuyển sang giai đoạn thanh toán, Website TMĐT sẽ yêu cầu khách hàng chọn “hình thức thanh toán”. Nếu chọn ví điện tử, Website sẽ gửi mật khẩu OTP qua tin nhắn tới số điện thoại di động đã khách hàng đăng ký(điều kiện bạn đã lưu ví điện tử trong website).
* Bước 3: Khách hàng nhập mật khẩu OTP đã nhận được để xác thực thông tin thanh toán.
* Bước 4: Sau khi nhập đúng mật khẩu OTP để xác thực thông tin thanh toán, khách hành sẽ được xác nhận là đã thanh toán thành công.

Qua các bước sử dụng mật khẩu OTP trong thanh toán trực tuyến cho thấy phương pháp xác thực này đảm bảo an toàn tốt hơn cho danh tính của khách hàng so với phương pháp xác thực với Username/Password, mật khẩu OTP trong mỗi phiên giao dịch là khác nhau (sử dụng một lần cho mỗi phiên giao dịch), với thời gian hợp lệ nhất định, tránh rủi ro khi tài khoản người dùng bị lộ nhưng sẽ không thực hiện thành công nếu không có được mã OTP.

Ngoài việc hỗ trợ xác thực trong thanh toán giao dịch ngân hàng, giao dịch thương mại điện tử, nó còn được sử dụng hỗ trợ xác thực đăng nhập sử dụng máy tính cá nhân và trong xác thực đăng nhập tài khoản thư điện tử,….

# **CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH ỨNG DỤNG MẬT KHẨU OTP CHO HỆ THỐNG HỌC TẬP**

## 1.  Đặt vấn đề

Sự phát triển đột phá của Công nghệ Thông tin (CNTT) trong thế kỷ 20 đã tạo nền tảng quan trọng cho những thành tựu khoa học công nghệ ở thế kỷ 21. Trong đó, đào tạo trực tuyến nổi lên như một cuộc cách mạng trong giáo dục, ứng dụng các giải pháp tiên tiến để truyền tải kiến thức và kỹ năng hiệu quả. Với sự hỗ trợ của máy tính và Internet, người học có thể tiếp cận các bài giảng mọi lúc, mọi nơi mà vẫn đạt hiệu quả cao.

Tại Việt Nam, đặc biệt trong giáo dục đại học và sau đại học, việc ứng dụng CNTT là yếu tố cần thiết để nâng cao chất lượng đào tạo, rút ngắn khoảng cách với các nước tiên tiến. Học tập trực tuyến cho phép sinh viên chủ động lập kế hoạch học tập, còn giáo viên có thể cập nhật nội dung thường xuyên và đánh giá mức độ tiếp thu của học viên thông qua hệ thống tự động.

Tuy nhiên, môi trường học tập trực tuyến thường xuyên kết nối Internet cũng tiềm ẩn nguy cơ về bảo mật, khi thông tin tài khoản và cá nhân của người dùng dễ bị tấn công. Đảm bảo an toàn thông tin cho học viên trở thành ưu tiên hàng đầu. Một giải pháp hiệu quả là áp dụng mật khẩu OTP, giúp tăng cường bảo mật và bảo vệ người dùng trong các hoạt động học trực tuyến.

## 2. Ứng dụng mật khẩu OTP cho hệ thống học tập trực tuyến.

**2.1  Thuật toán TOTP:**

Độ an toàn của mã OTP phụ thuộc tính bảo mật của hàm băm. Tất cả các hệ thống sử dụng OTP phải hỗ trợ MD5 nên hỗ trợ SHA và có thể hỗ trợ MD4.

Khi xây dựng ứng dụng sinh mật khẩu OTP cho hệ thống học tập trực tuyến em tim hiểu và sử dụng thuật toán TOTP là một thuật toán sinh mật khẩu OTP dựa trên hàm hash SHA-1. Thuật toán này đã được trình bày ở chương 2 (Mục 3.2). Trong ứng dụng này thì máy tính của người học viên tham gia học tập trực tuyến và server cùng sử dụng hàm băm an toàn SHA-1 cho việc sinh và xác thực mã OTP.

Bên máy tính cá nhân mã OTP được tạo ra với các tham số: (thời gian thực và username/password).

Bên phía server cũng thực hiện băm với đầu vào là thời gian thực, username/password của từng người dùng để sinh mã OTP. Kết quả này được đem so sánh với mã OTP mà người dùng nhập vào để đưa ra quyết định.

Các tham số đầu vào để tạo mã OTP của hệ thống:

* Username/password: Dạng chuỗi.
* Thời gian thực: Thời gian bao gồm: Năm, tháng, ngày, giờ, phút.

Các tham số này được kết nối với nhau và đưa qua hàm băm SHA-1. Đầu ra sẽ được chuyển về dạng dễ sử dụng cho người dùng gồm 6 ký tự.

**2.2. Ứng dụng OTP cho việc xác thực tài khoản trong học tập trực tuyến**

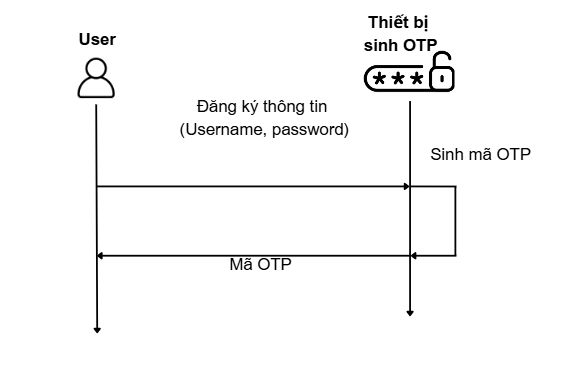
Hệ thống bao gồm: học viên, web server và thiết bị của người sử dụng được dùng để tạo mã OTP (ở đây là máy tính cá nhân đã cài đặt chương trình có khả năng sinh mã OTP và đã được đồng bộ thời gian với server ).

Để thực hiện việc xác thực hai yếu tố Người học viên phải đăng kí tài khoản với nhà cung cấp dịch vụ (server). Server cung cấp cho người sử dụng thông tin tài khoản để xác thực trên hệ thống. Những thông tin này sẽ được lưu vào cơ sở dữ liệu trên server.



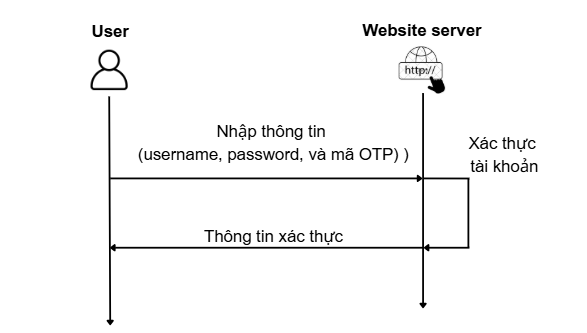
*Hình 3.1 Quá trình đăng ký*

Người sử dụng khi cần xác thực trên trang web. Sẽ chạy chương trình ứng dụng trên thiết bị có khả năng sinh mã OTP ( Ở đây là máy tính cá nhân) của họ. Sau đó nhập những thông tin mà chương trình yêu cầu để lấy mã OTP.



*Hình 3.2. Quá trình trình sinh mã OTP*

Sau khi lấy được mã OTP. Học viên sẽ nhập vào trang web cùng với usename và password mà họ đã đăng kí với nhà cung cấp dịch vụ. Server sau khi nhận được thông tin xác thực của khách hàng sẽ tiến hành kiểm tra sự hợp lệ của thông tin và trả lại kết quả xác thực cho khách hàng. Quá trình này được mô tả như hình 3.3.



*Hình 3.3. Quá trình xác thực mã OTP*

## 3. Xây dựng chương trình ứng dụng sinh OTP cho hoạt động học tập trực tuyến.

**3.1. Mô tả hoạt động của hệ thống học tập trực tuyến sử dụng mật khẩu OTP**



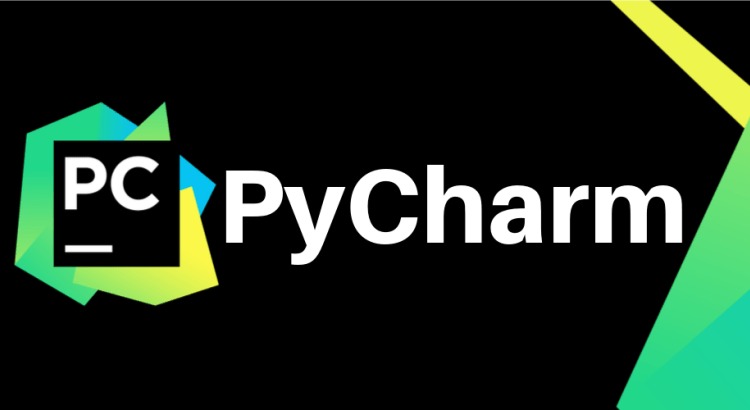
*Hình 3.4. Mô hình sử dụng OTP để xác thực tài khoản trong hệ thống học trực tuyến*

Để mô phỏng ứng dụng OTP cho việc xác thực thông tin tài khoản trong hoạt động học tập trực tuyến, em xây dựng mô hình dựa trên cơ chế sinh mật khẩu OTP ở cả hai phía và áp dụng thuật toán TOTP.

Mô hình sinh mật khẩu OTP và xác thực áp dụng trong hoạt động học tập trực tuyến được mô tả như trên hình 3.4

**3.2.  Ứng dụng và thư viện được sử dụng để cài đặt chương trình**

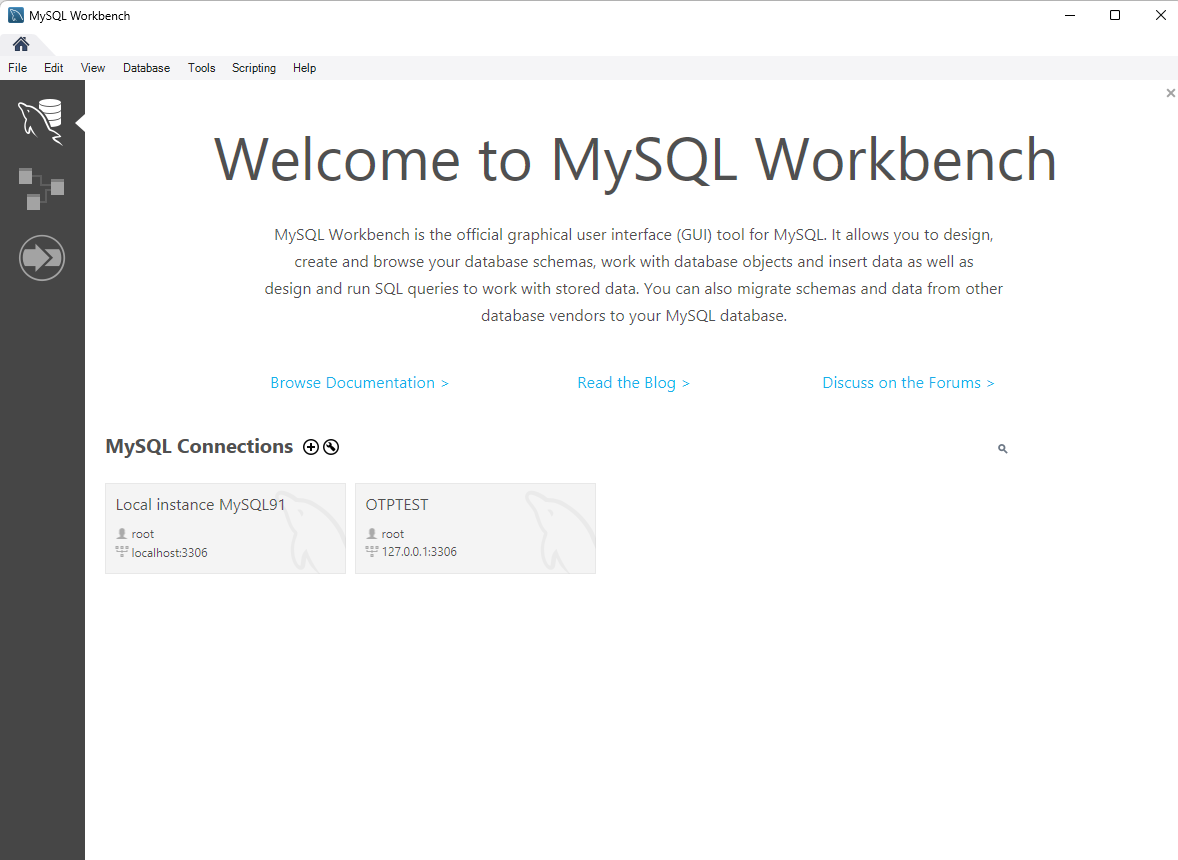
Phần mềm bên server được xây dựng dưới dạng webserver. Phía server dựa vào thông tin tài khoản và mã OTP.



PyCharm - Công cụ phát triển Python PyCharm là một môi trường phát triển tích hợp (IDE) mạnh mẽ dành cho Python, được phát triển bởi JetBrains. Đây là một công cụ tuyệt vời cho các nhà phát triển web, đặc biệt khi sử dụng các framework như Flask hoặc Django để xây dựng các ứng dụng web.

Tính năng nổi bật của PyCharm:

* Hỗ trợ quản lý môi trường ảo (virtual environment) để làm việc với các gói Python.
* Trình soạn thảo mã mạnh mẽ với tính năng tự động hoàn thiện, kiểm tra lỗi, và refactor.
* Hỗ trợ tích hợp với Git, giúp dễ dàng quản lý và đồng bộ mã nguồn.
* Debugging trực quan giúp kiểm tra và sửa lỗi hiệu quả.



MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến. Với trang web đăng nhập và xác thực mã OTP, MySQL thường đượ sử dụng để lưu trữ và quản lý thông tin người dùng, bao gồm:

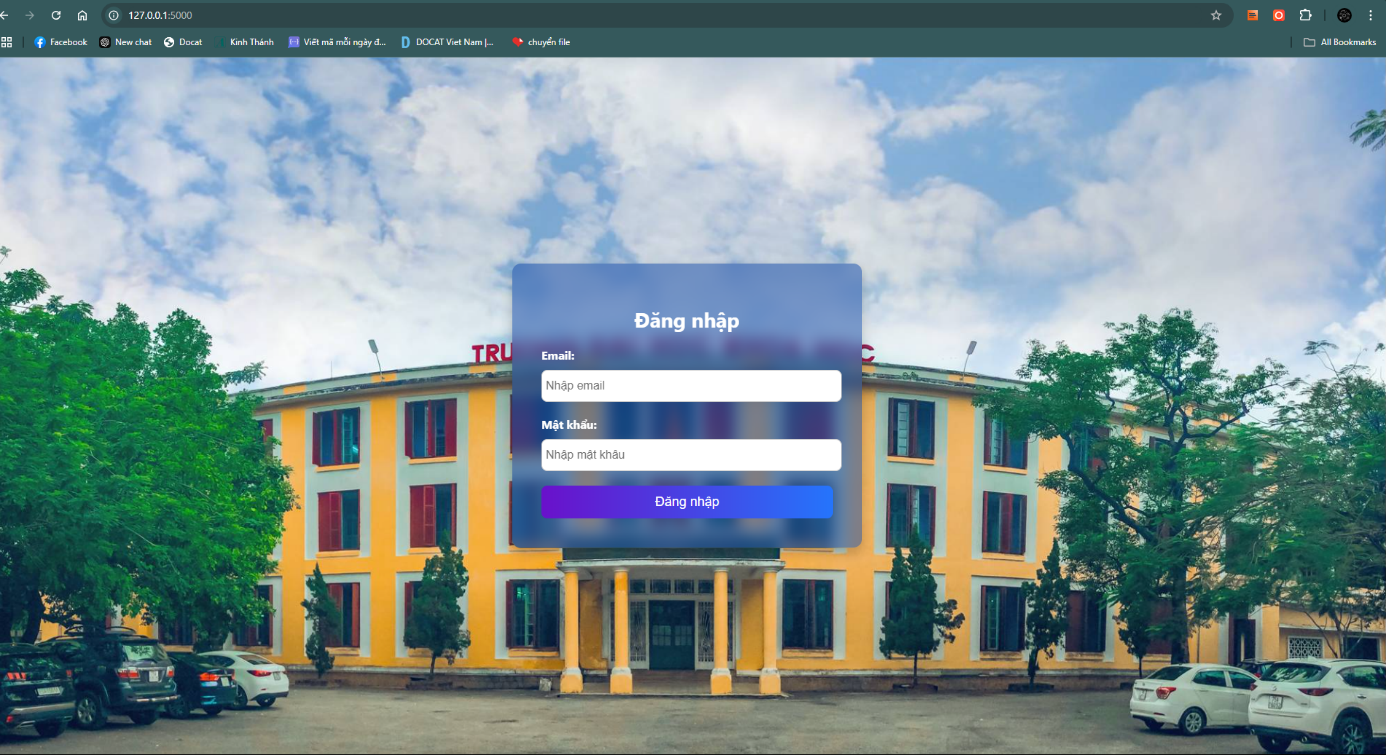
* Thông tin tài khoản (email, username, mật khẩu được mã hóa).
* Mã OTP (mã OTP tạm thời được sinh ra và lưu trữ để xác thực).
* Thời gian hết hạn OTP (đảm bảo mã OTP chỉ hợp lệ trong khoảng thời gian nhất định).

**3.3. Các bước cài đặt và chạy chương trình**

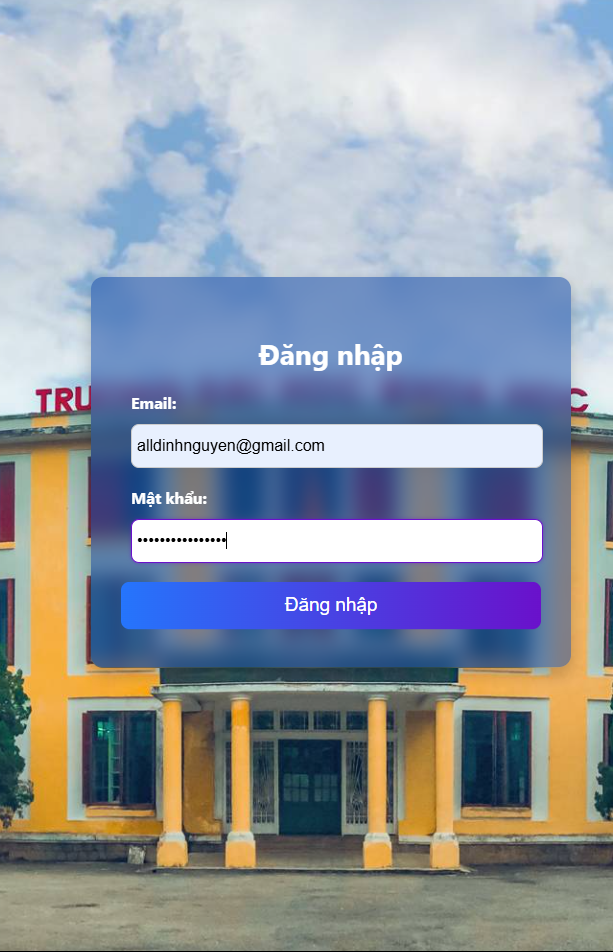
**3.3.1. Thiết lập môi trường Server cho OTP**

* + Tạo môi trường ảo trong PyCharm (hoặc CMD/PowerShell). Gõ lệnh sau để tạo môi trường ảo:python -m venv .venv
  + Kích hoạt môi trường ảo: venv\Scripts\activate
  + Cài đặt các thư viện cần thiết
* Trong Terminal (đang kích hoạt môi trường ảo), chạy lệnh sau: pip install flask mysql-connector-python flask-session email-validator
* Flask: Framework để xây dựng ứng dụng web.
* mysql-connector-python: Để kết nối MySQL với Python.
* flask-session: Để quản lý session người dùng trong Flask.
* email-validator: Để kiểm tra định dạng email hợp lệ.
* Tạo cơ sở dữ liệu mới và lưu thông tin của user và password để truy vấn vào trong dự án

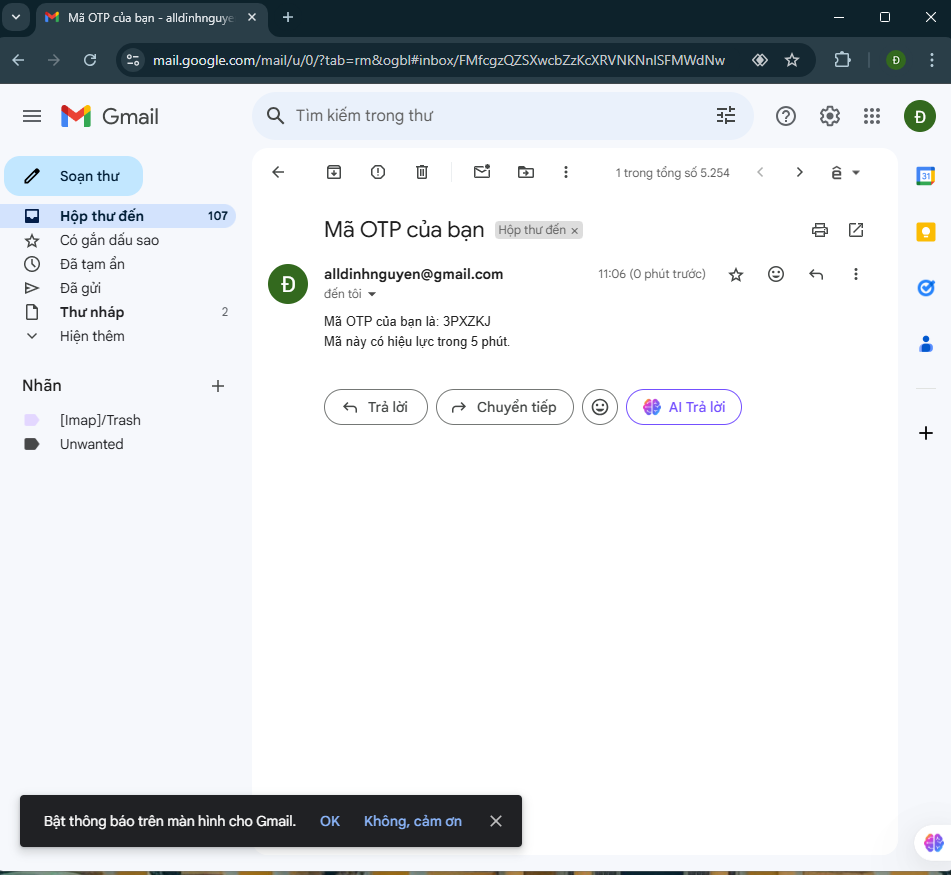
**3.3.2  Kết quả thử nghiệm**



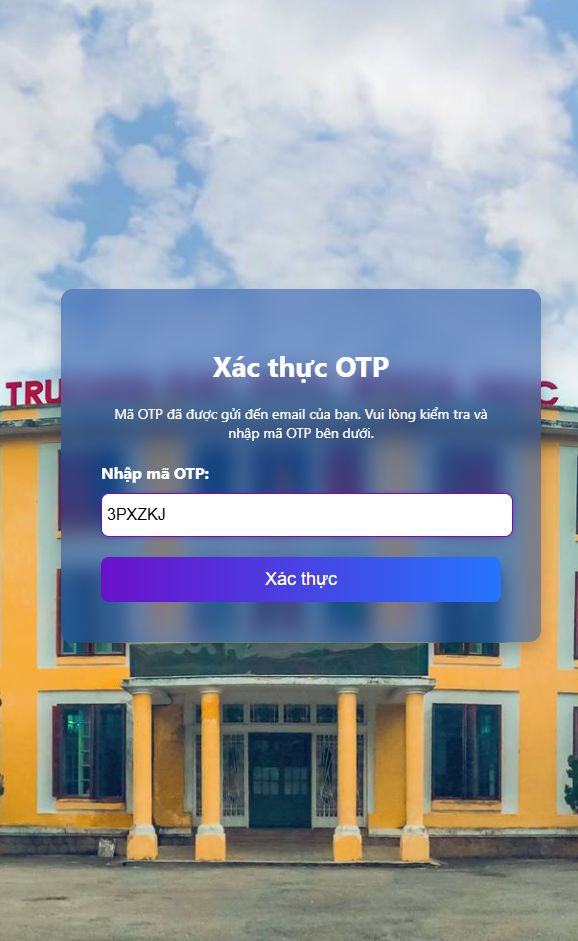
Nhập thông tin về user, pass. Thông tin này được gửi đến server và được CSDL ở server lưu trữ (ở đây sử dụng MySQl để tạo dữ liệu cho dự án) , sau đó yêu cầu cấp mã OTP để đăng nhập.



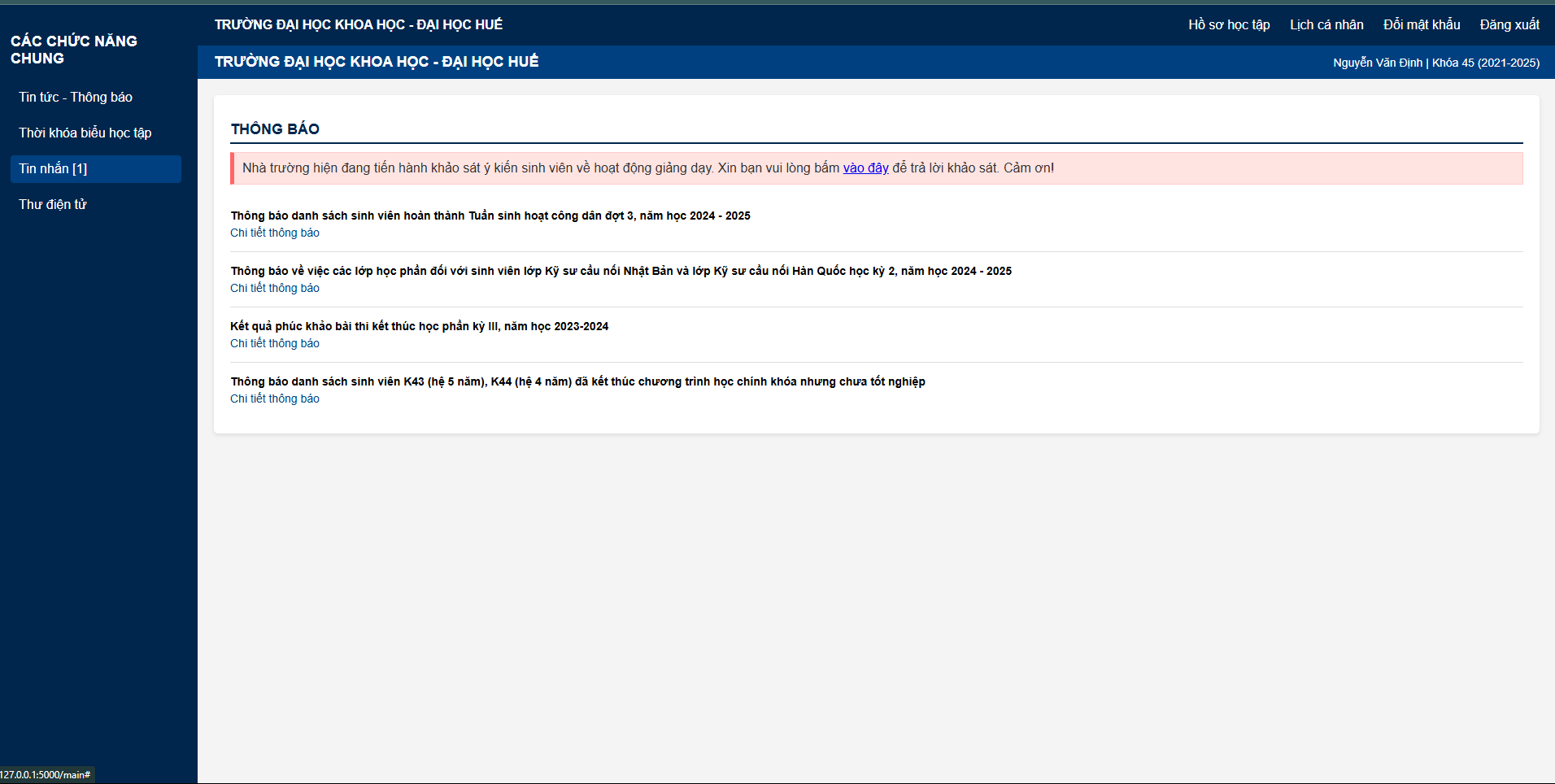
Sau khi nhấn nút đăng nhập sẽ kiểm tra tính đúng sai của email và password nếu đã nhập đúng mã OTP (mã 6 chữ số) sẽ được cấp cho bạn theo thuật toán TOTP và có thời hạn sử dụng là 1 phút



Khi đó mã sẽ được gửi về gmail bạn đã nhập

**

Nếu nhập sai mã OTP đã cung cấp thì sẽ không thể hoàn thành bước đăng nhập.



Sau khi đã nhập đúng mã OTP hệ thống đã tự cấp thì bạn sẽ đăng nhập được vào trang chính.

Tất cả các thành phần cấu trúc dự án đều có trên github.com qua đường link (https://github.com/concasau12/X-c-th-c-y-u-t-th-2)

**3.4. Đánh giá**

Về độ an toàn: Các mật khẩu otp được sinh ra an toàn khó đoán trước được.

Khả năng ứng dụng của hệ thống: Hiện nay hệ thống có thể áp dụng cho một website học tập trực tuyến. Cách thức hoạt động có thể đảm bảo an toàn cho người học về thông tin cá nhân, hoạt động học tập, nội dung học tập.

Hạn chế của hệ thống: Thuật toán sinh mật khẩu otp sử dụng trong chương trình chỉ sinh ra một số lượng otp giới hạn. Nghĩa là khi kho mật khẩu otp được sử dụng hết sẽ sảy ra sự lặp lại của mật khẩu. Bên cạnh đó mô hình sinh mật khẩu otp ở đây có hạn chế về tính linh hoạt trong sử dụng. Nghĩa là một người muốn sử dụng hệ thống này thì nhất định là phải có máy tính cá nhân, các thiết bị di động muốn hỗ trợ được thì phải cài đặt một số phần mềm tương đối khó khăn.

Mật khẩu OTP được sinh ra theo thuật toán TOTP đảm bảo được độ an toàn về thông tin. Tuy nhiên, trong thực tế hệ thống học tập trực tuyến đã được trình bày muốn được đưa vào sử dụng và đảm bảo tính toàn vẹn về thông tin cần phải kết hợp với một số phương pháp xác thực khác. (Ví dụ như: nhận diện hình ảnh trong trường hợp học viên nhờ người “học hộ”).

**KẾT LUẬN**

## 1.  Kết luận

Phương pháp xác thực bằng mật khẩu OTP mang lại nhiều sự an toàn thông tin cho người sử dụng, và hiện phương pháp này đang được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau.

Bằng cách sử dụng một số thuật toán mật mã khó đoán trước nên sự tấn công của các hacker, mã độc được giảm thiểu.

Tuy nhiên các chuyên gia cũng khuyến cáo rằng dù phương thức bảo mật nào rồi cũng sẽ có các lỗ hổng và sẽ bị tấn công. Do đó người dùng cần phải cảnh giác với những thay đổi bất thường về thông tin của mình.

## 2.  Kiến nghị

Chương trình sau khi hoàn thiện và đưa vào thực tế hoạt động có thể giúp đảm bảo an toàn thông tin cho người học tập trực tuyến. Tuy nhiên để hệ thống hoạt động tốt hơn thì cần phải kết hợp với một số phương pháp xác thực khác để đảm bảo tính toàn vẹn thông tin.

## 3.  Hướng phát triển của đề tài

Sau khi thực hiện chương trình với kỹ thuật sinh mật khẩu otp ở cả hai phía thì về lý thuyết cơ chế sinh mật khẩu này là rất an toàn, tuy nhiên trên thực tế để ứng dụng xác thực thông tin người dùng trong các hoạt động trực tuyến thì cần kết hợp với một số hình thức xác thực khác để đảm bảo độ an toàn cao nhất cho thông tin, vì vậy hướng nghiên cứu tiếp theo của đề tài là nghiên cứu một số phương pháp xác thực có độ an toàn cao cho thông tin như sinh trắc học.

# **TÀI LIỆU THAM KHẢO**

## Tài liệu tiếng việt

[1] Nguyễn Bình, Trần Đức Sự (2011), *Cơ sở lý thuyết mật mã*, Nxb Bộ thông tin và truyền thông, Hà nội.

[2] Nguyễn Ngọc Cương, Trần Thị Lượng (2013), *Giáo trình Mật mã ứng dụng trong an toàn thông tin*, Nxb Học viện Kỹ thuật mật mã, Hà nội.

[3] Trần Đức Sự, Nguyễn Văn Tảo, Trần Thị Lượng (2015), *An toàn bảo mật dữ liệu*, Nxb Đại học Thái Nguyên, Thái Nguyên.

[4] Nguyễn Vạn Phúc, Lê Trọng Hiệp (2011), "Bảo mật trong giao dịch sử dụng công nghệ xác thực OTP"*,* Tạp chí An toàn thông tin tháng 7/2011.

[5] Đặng Mạnh Phổ (2011), "Ứng dụng xác thực đa yếu tố trong giao dịch ngân hàng điện tử", Tạp chí An toàn thông tin tháng 7/2011.

## Tài liệu tiếng anh

[6] Behrouz  A.Forouzan  (2007),  *Cryptography  and  Network  security* (Chapter 14), McGraw Hill.

[7] M. Matsumoto and M. Saito (2006), “*SIMD-oriented fast Mersenne twister: a 128-bit pseudorandom number generator*” in Proceeding of MCQMC.