**ĐẠI HỌC HUẾ**
**TRƯỜNG ĐẠI HỌC KHOA HỌC**
**KHOA CÔNG NGHỆ THÔNG TIN**
**AN NINH MẠNG**

# ĐỀ TÀI:
# TÌM HIỂU GIẢI PHÁP ONE-TIMEPASSWORD VÀ XÂY DỰNG ỨNG DỤNG THỬ NGHIỆM


Sinh viên thực hiện:
Thái Quang Duy Anh (Nhóm trưởng)
Võ Văn Bin
Nguyễn Thành Danh
Nguyễn Ngọc Vĩ
Phan Văn Quyền

# I.DANH MỤC TỪ VIẾT TẮT

1.OTP :(One-Time Password) OTP là mật khẩu dùng một lần, được sinh ra để sử dụng chỉ cho một lần xác thực và có hiệu lực trong thời gian rất ngắn hoặc một phiên duy nhất.

2. 2FA :(Two-Factor Authentication) 2FA là xác thực hai yếu tố, yêu cầu người dùng chứng minh danh tính bằng hai loại yếu tố khác nhau.

3. MFA :(Multi-Factor Authentication) MFA là xác thực đa yếu tố, mở rộng của 2FA, sử dụng từ 2 yếu tố trở lên

4. HOTP :(HMAC-Based One-Time Password) HOTP là thuật toán sinh OTP dựa trên bộ đếm (counter), được chuẩn hóa bởi RFC 4226.

5. HMAC :(Hash-based Message Authentication Code) HMAC là cơ chế xác thực thông điệp bằng hàm băm và khóa bí mật.

II.MỤC LỤC

## MỞ ĐẦU

Trong bối cảnh công nghệ thông tin ngày càng phát triển mạnh mẽ, các hệ thống thông tin và dịch vụ trực tuyến đã trở thành một phần không thể thiếu trong đời sống xã hội. Tuy nhiên, song song với sự tiện lợi đó là những thách thức ngày càng nghiêm trọng về an toàn và bảo mật thông tin. Các hình thức tấn công như đánh cắp mật khẩu, nghe lén, giả mạo danh tính hay tấn công trung gian đang ngày càng tinh vi, đòi hỏi các giải pháp xác thực người dùng có mức độ an toàn cao hơn so với phương thức mật khẩu tĩnh truyền thống.

One-Time Password (OTP) là một trong những giải pháp xác thực hiệu quả nhằm nâng cao mức độ bảo mật cho các hệ thống thông tin. OTP là mật khẩu chỉ có hiệu lực trong một lần đăng nhập hoặc trong một khoảng thời gian ngắn, giúp giảm thiểu nguy cơ bị đánh cắp và tái sử dụng mật khẩu. Hiện nay, OTP được ứng dụng rộng rãi trong nhiều lĩnh vực như ngân hàng điện tử, thương mại điện tử, hệ thống quản lý doanh nghiệp và các dịch vụ trực tuyến yêu cầu tính bảo mật cao.

Xuất phát từ nhu cầu thực tiễn đó, đề tài **"Tìm hiểu giải pháp One-Time Password và xây dựng ứng dụng thử nghiệm"** được thực hiện nhằm nghiên cứu tổng quan về cơ chế hoạt động của OTP, các phương pháp sinh và xác thực OTP phổ biến, cũng như đánh giá ưu nhược điểm của từng phương pháp. Bên cạnh đó, đề tài tập trung vào việc xây dựng một ứng dụng thử nghiệm áp dụng cơ chế OTP trong xác thực người dùng, qua đó giúp làm rõ quy trình triển khai OTP trong thực tế và đánh giá tính khả thi của giải pháp.

Thông qua việc nghiên cứu lý thuyết kết hợp với xây dựng ứng dụng thực nghiệm, đề tài hướng đến mục tiêu nâng cao hiểu biết về các giải pháp xác thực an toàn, đồng thời góp phần đề xuất một hướng tiếp cận hiệu quả trong việc tăng cường bảo mật cho các hệ thống thông tin hiện nay.

## CHƯƠNG 1: Tổng quan về OTP

### 1.1. Định nghĩa bản chất của OTP

- OTP (One-Time-Password) được gọi là mật khẩu dùng một lần trong quá trình xác thực người dùng. OTP được tạo ra bởi hệ thống một phiên bản xác thực riêng biệt hoặc trong một khoảng thời gian rất ngắn. Khác với mật khẩu tĩnh, mật khẩu này không thể tái sử dụng sau khi đã hết hạn [1].

- Mục tiêu của là tăng cường mức độ bảo mật trong quá trình xác thực người dùng, hạn chế các rủi ro đến việc đánh cắp mật khẩu như nghe lén, dò mật khẩu, tấn công từ điển hoặc tấn công brute-force [1].

### 1.2. Vai trò của OTP trong hệ thống xác thực

Mã OTP đóng vai trò cực kỳ quan trọng trong việc bảo mật các dịch vụ giao dịch ngân hàng, xác minh tài khoản người dùng trực tuyến [2]. Dưới đây là những lợi ích của mã OTP mang lại cho người dùng :

- Xác nhận giao dịch: đây là bước quan trọng để đảm chỉ có chủ tài khoản có thể thực hiện các giao dịch thanh toán trực tuyến như chuyển tiền, thanh toán hóa đơn, mua sắm online,…[3].

- Chống lừa đảo: khi nhập mật khẩu vào một trang website giả mạo, kẻ gian cũng không thể hoàn tất nếu không có mã OTP được gửi về số điện thoại của người đó [3].

- Bảo vệ tài khoản người dùng khi đăng nhập trên thiết bị lạ: khi đăng nhập trên một thiết bị mới, nhiều hệ thống sẽ yêu cầu nhập thêm mã OTP để xác minh được người dùng đó [3].

### 1.3. Các hình thức triển khai OTP

- OTP qua SMS (SMS-based OTP): là tin nhắn của hệ thống sẽ gửi về điện thoại để nhập mã xác thực, nó có thể là một dãy số hoặc một chuỗi kết hợp chữ số và chữ cái. OTP được tạo ra chỉ một lần và sau đó không còn tác dụng. Thời gian có hiệu lực của mật khẩu OTP rất ngắn, nó sẽ bị vô hiệu khi hết thời gian và được thay thế bởi một mật mã mới.[4]

- Voice OTP: là một phương thức xác thực danh tính được gọi đến người dùng thông qua cuộc gọi thay vì gửi tin nhắn. Khi cần xác thực, hệ thống sẽ gọi điện và đọc to mã OTP để người dùng nhập vào hệ thống. Đây là một phương thức xác thực ổn định hơn so với SMS OTP, tránh được các vấn đề như tin nhắn bị chậm và độ bảo mật cao hơn để hạn chế các tin nhắn giả mạo. [5]

- Email OTP: đây là một hình thức xác thực hệ thống gửi mã OTP qua địa chỉ email đến với nhiều người dùng cùng một lúc để đảm bảo chỉ có người dùng họ mới có thể truy cập [6].

- Soft Token: là token xác thực dựa trên phần mềm chạy trên thiết bị của người dùng để bảo vệ tài khoản cũng như thông tin cá nhân của của người dùng. Để truy cập vào tài khoản cũng như thực hiện giao dịch, người dùng bắt buộc phải nhập mã xác thực khi được hệ thống gửi về. Để tăng tính bảo mật và độ chính xác, người dùng có thể cần nhập kết hợp mật khẩu và mã token. [7]

- Hard Token: là thiết bị vật lý được sử dụng để đưa ra mã thông báo cho người dùng khi truy cập vào mạng hay hệ thống, từ đó giúp xác định danh tính trong hệ thống bảo mật. So với mật khẩu truyền thống, phương pháp này có độ bảo mật cao hơn. [7]

# CHƯƠNG 2: MÔ HÌNH XÁC THỰC VÀ PHÂN TÍCH THUẬT TOÁN HOTP VÀ TOTP

## 2.1. Mô hình xác thực OTP tổng quát

- Mô hình xác thực OTP tổng quát là mô tả toàn bộ quy trình + thành phần tham gia trong việc xác minh danh tính người dùng bằng mật khẩu dùng một lần (OTP). Nghĩa là thay vì dùng mật khẩu cứng lâu dài, hệ thống tạo ra một mã chỉ dùng một lần và có hiệu lực trong thời gian ngắn để xác thực mỗi lần đăng nhập hoặc giao dịch [8].

- OTP thường là một chuỗi số ngẫu nhiên hoặc một dãy ký tự được hệ thống gửi tới người dùng qua SMS, email, voice hoặc ứng dụng xác thực và chỉ có hiệu lực một lần duy nhất trong một khoảng thời gian ngắn – hết thời gian là vô hiệu.

- Các thành phần trong mô hình xác thực OTP:

+ Người dùng: thực hiện yêu cầu đăng nhập và giao dịch.

+ Thiết bị nhận OTP: điện thoại di động, email người dùng hoặc thiết bị token.

+ Máy chủ xác thực (Authentication Server): sinh và kiểm tra mã OTP.

+ Kênh truyền OTP: SMS, email, voice hoặc ứng dụng xác thực

- Quy trình xác thực OTP tổng quát:

1. Người dùng nhập thông tin đăng nhập (tên người dùng và mật khẩu)
2. Hệ thống yêu cầu xác thực OTP.
3. Máy chủ sinh OTP và gửi đến tài khoản người dùng.
4. Người dùng nhập mã OTP vào hệ thống
5. Máy chủ kiểm tra hợp lệ của OTP và quyết định cho phép hoặc từ chối truy cập.

Mô hình này được xác kết hợp hai yếu tố (2FA) hoặc xác thực đa yếu tố (MFA) để nâng cao mức độ bảo mật của hệ thống.

2.2. Thuật toán HOTP (HMAC-Based One-Time Password)

2.2.1. Khái niệm thuật toán HOTP

- Thuật toán HOTP là một thuật toán sinh mật khẩu dùng một lần dựa trên chuẩn RFC 4226. Thuật toán này sử dụng hàm băm HMAC kết hợp với một khóa bí mật dùng chung và một bộ đếm tăng dần để tạo mã OTP duy nhất cho mỗi lần xác thực.

2.2.2. Nguyên lý hoạt động của thuật toán HOTP

Thuật toán HOTP được mô tả bằng công thức:

$$HOTP(K,C) = Truncate(HMAC\_SHA\text{-}1(K,C)) \quad (1)$$

Trong đó:

+ K: là giá trị chia sẻ bí mật giữa Người dùng và máy chủ

+ C: là bộ đếm được tăng lên sau mỗi lần sinh OTP.

+ Truncate là hàm cắt kết quả để lấy ra một chuỗi số có độ dài cố định (thường là 6 hoặc 8 chữ số)

2.2.3. Ưu điểm của thuật toán HOTP

+ Linh hoạt về thời gian: HOTP sinh OTP dựa trên bộ đếm (counter), không cần đồng bộ thời gian giữa Client và Server. Điều này giúp hệ thống tránh lỗi lệch giờ, nhất là mấy môi trường Server cũ hoặc thiết bị người dùng lệch múi giờ [9].

+ Thuật toán đơn giản, dễ triển khai: thuật toán HOTP chỉ cần một khóa bí mật dùng chung (secret key), một bộ đếm tăng dần và hàm băm HMAC (thường là HMAC-SHA-1). Điều này giúp thuật toán code gọn, dễ debug, dễ tích hợp vào hệ thống xác thực hiện có [9].

+ Độ bảo mật cao: mỗi mã OTP chỉ sử dụng đúng một lần, không thể sử dụng cho lần tiếp theo. Dựa trên HMAC nên rất khó bị giả tạo nếu khóa bí mật được bảo vệ tốt [9].

+ Khả năng tương thích: HOTP có thể dễ dàng tích hợp vào các hệ thống xác thực hiện có, mang lại sự nâng cao về bảo mật và khả năng tương thích ngược [9].

2.2.4. Nhược điểm của thuật toán HOTP

- HOTP có hiệu lực lâu hơn TOTP: mức tăng của HOTP chỉ thay đổi sau khi xác thực thành công. Điều này nếu người dùng không truy cập tài khoản của họ trong một thời gian dài, mã OTP hiện tại sẽ vẫn có hiệu lực trong thời gian đó. [10]

- HOTP có thể gặp sự cố đồng bộ hóa: Bộ đếm sự kiện trong HOTP tạo ra khả năng mất đồng bộ giữa máy chủ và mã OTP. Ví dụ, nếu nút trên mã OTP được nhấn quá nhiều lần, giá trị được máy chủ tính toán sẽ không khớp với giá trị hiển thị trên màn hình mã. Để khắc phục điều này, máy chủ cần có khả năng chấp nhận các mã OTP trước và sau đó, và tất cả các mã OTP được chấp nhận sẽ tạo ra một cửa sổ xác thực. Cửa sổ xác thực rộng hơn này làm tăng nguy cơ kẻ tấn công xâm nhập vào tài khoản người dùng bằng cách tấn công vét cạn tất cả các giá trị OTP tiềm năng. [10]

2.3. Thuật toán TOTP

2.3.1. Khái niệm thuật toán TOTP

- TOTP là một mã tạm thời, chỉ được phép sử dụng phù hợp với thời gian hiện tại bằng một thuật toán xác thực người dùng. TOTP là một lớp bảo mật cung cấp cho tài khoản dựa vào xác thực hai yếu tố (2FA) và đa yếu tố (MFA). Nghĩa là sau khi đăng nhập, hệ

thống sẽ yêu cầu người dùng nhập một mã được gửi về trong một khoảng thời gian ngắn. [13]

2.3.2. Nguyên lý hoạt động của thuật toán TOTP

- Thuật toán TOTP được mô tả bằng công thức:

TOTP(K,T) = HOTP(K,T)

Trong đó:

+ K là khóa bí mật được chia sẻ giữa người dùng và máy chủ xác thực

+ T là bộ đếm thời gian, được tính bằng công thức:

T = (CurrentTime – T0) / X

+ CurrentTime: thời gian hiện tại

+ X: thời gian hiệu lực của OTP (thường là 30 giây)

Thuật toán TOTP hoạt động dựa trên 3 nguyên lý chính:

1. Phụ thuộc thời gian:

- OTP được sinh ra từ thời gian hệ thống, do đó:
- OTP tự động thay đổi sau mỗi khoảng X giây
- OTP cũ sẽ hết hạn và không thể sử dụng lại

2. Dùng hàm băm một chiều (HMAC):

- Khóa bí mật được kết hợp với thời gian thông qua hàm HMAC-SHA1
- Không thể suy ngược lại khóa bí mật từ OTP

3. Không cần lưu OTP:

- Hệ thống không lưu OTP
- Khi xác thực, OTP được sinh lại từ cùng khóa bí mật và thời gian hiện tại

2.3.3. Ưu điểm của thuật toán TOTP

- Bảo mật cao: thuật toán TOTP hiệu quả hơn HOTP vì chỉ có hiệu lực trong thời gian ngắn. Ngay cả khi ai đó chặn mật khẩu, họ cũng không thể sử dụng nó sau khi thời gian giới hạn hết hạn. [10]

- Hơn nữa, mỗi mã TOTP đều là duy nhất, giảm thiểu rủi ro trùng lặp. TOTP tăng cường bảo mật trong các hệ thống xác thực đa yếu tố, khiến tội phạm khó xâm nhập tài khoản hơn ngay cả khi chúng có thông tin đăng nhập cơ bản của người dùng. [10]

- TOTP khuyến khích người dùng xác thực các giao dịch nhanh chóng, từ đó tăng hiệu quả hoạt động. [10]

2.3.4. Nhược điểm của thuật toán TOTP

- "Người dùng cần nhập mật khẩu vào trang xác thực, điều này có thể làm tăng nguy cơ bị tấn công lừa đảo. Kẻ tấn công có thể giả mạo các trang web này và lừa người dùng tiết lộ mật khẩu dùng một lần của họ"[11].

- "TOTP dựa trên một mã bí mật được cả máy khách và máy chủ cùng biết. Điều này tạo ra nhiều điểm tiềm ẩn hơn về khả năng bị đánh cắp mã bí mật. Nếu kẻ tấn công có được quyền truy cập vào mã bí mật này, chúng có thể tạo ra các mã TOTP hợp lệ mới tùy ý, điều này đặc biệt nguy hiểm nếu cơ sở dữ liệu xác thực lớn bị xâm phạm"[11].

- "Thuật toán TOTP phụ thuộc vào việc đồng bộ hóa thời gian chính xác giữa bộ tạo mã (thường là thiết bị phần cứng hoặc ứng dụng phần mềm) và máy chủ. Sai lệch trong cài đặt thời gian có thể dẫn đến việc mã được tạo ra không khớp với mã OTP mà máy chủ mong đợi, khiến nó trở nên vô dụng"[11].

## CHƯƠNG 3: Thuật toán sinh TOTP

3.1. Mục đích của thuật toán TOTP

- Thuật toán TOTP (Time-based One-Time Password) được sử dụng để sinh mật khẩu dùng một lần dựa trên thời gian hiện tại, nhằm tăng cường bảo mật trong xác thực người dùng.

- Mỗi mã OTP chỉ có hiệu lực trong một khoảng thời gian ngắn (thường là 30 giây) và không thể sử dụng lại.

3.2 Đầu vào và đầu ra của thuật toán

3.2.1. Đầu vào (Input):

- Khóa bí mật (Secret Key – K)
- Là chuỗi ký tự chỉ được biết bởi hệ thống và người dùng.
- Thời gian hiện tại (Unix Time – t)
- Tính bằng số giây kể từ 01/01/1970.

3.2.2. Đầu ra (Output):

Mã OTP gồm n chữ số (thường là 6 chữ số)

3.3. Các tham số của thuật toán

- X = 30: Khoảng thời gian hiệu lực của OTP (giây)
- T0 = 0: Thời điểm bắt đầu (theo chuẩn RFC 6238)
- Digits = 6: Số chữ số của OTP
- Hash = HMAC - SHA1: Thuật toán băm dùng trong TOTP

3.4. Mô tả thuật toán dạng OTP

Thuật toán: SINH MÃ OTP THEO TOTP

Bước 1: Lấy thời gian hiện tại

t ← số giây hiện tại kể từ 01/01/1970

Bước 2: Tính bộ đếm thời gian

T ← (t − T0) / X

*Trong đó:*

- *T0 = 0*

- *X = 30*

*Bộ đếm T chỉ thay đổi sau mỗi 30 giây.*

Bước 3: Chuyển T thành mảng byte

T → mảng 8 byte (big-endian)

Bước 4: Tính giá trị HMAC

H ← HMAC-SHA1(K, T)

H ← HMAC-SHA1(K, T)

- K: khóa bí mật
- H: chuỗi byte kết quả băm

Bước 5: Cắt động (Dynamic Truncation)

1. Lấy 4 bit cuối của byte cuối trong H:

offset ← H[last] & 0x0F

2. Lấy 4 byte liên tiếp từ vị trí offset
3. Ghép thành số nguyên không dấu 31 bit

Bước 6: Tạo mã OTP

OTP ← (binaryCode mod 10^Digits)

→ Đảm bảo OTP có đúng Digits chữ số

Bước 7: Xuất OTP

- Thêm số 0 phía trước nếu cần
- Trả về OTP cho người dùng

3.5. Thuật toán xác thực OTP

Thuật toán: KIỂM TRA OTP

Bước 1: Nhận OTP do người dùng nhập.

Bước 2: Sinh OTP tại thời điểm hiện tại bằng cùng khóa bí mật.

Bước 3: So sánh:

Nếu OTP_hệ_thống = OTP_người_dùng → HỢP LỆ

Ngược lại → KHÔNG HỢP LỆ

3.6. Điều kiện hoạt động đúng

Thuật toán TOTP hoạt động chính xác khi thỏa mãn các điều kiện sau:

- Khóa bí mật phải giống nhau giữa bên sinh OTP và bên xác thực
- Thời gian hệ thống phải đồng bộ
- OTP chỉ hợp lệ trong khoảng thời gian X giây
- OTP chỉ được sử dụng một lần

3.7 PHẦN CODE:

```
<!DOCTYPE html>
<html lang="vi">
<head>
<meta charset="UTF-8">
<title>TOTP Demo</title>
</head>
<body>

<h3>TOTP Generator & Verifier</h3>

<input id="secret" value="MY_SECRET_KEY">
```

```
<button onclick="genOTP()">Sinh OTP</button>
<p id="otp"></p>

<input id="userOtp" placeholder="Nhập OTP">
<button onclick="verifyOTP()">Xác thực</button>
<p id="result"></p>

<script>
const STEP = 30;      // Thời gian hiệu lực OTP (giây)
const DIGITS = 6;     // Số chữ số OTP

// Chuyển chuỗi → byte
const enc = s => new TextEncoder().encode(s);

// ================= SINH OTP =================
async function genOTP() {
  const key = await crypto.subtle.importKey(
    "raw", enc(secret.value),
    { name: "HMAC", hash: "SHA-1" },
    false, ["sign"]
  );

  const T = Math.floor(Date.now() / 1000 / STEP); // Bộ đếm thời gian
  const buf = new ArrayBuffer(8);
```

```
  new DataView(buf).setBigInt64(0, BigInt(T));

  const hmac = new Uint8Array(await crypto.subtle.sign("HMAC", key, buf));

  const o = hmac[hmac.length - 1] & 0xf; // offset cắt động
  const bin =
    ((hmac[o] & 0x7f) << 24) |
    ((hmac[o+1] & 0xff) << 16) |
    ((hmac[o+2] & 0xff) << 8) |
    (hmac[o+3] & 0xff);

  const otp = (bin % 10 ** DIGITS).toString().padStart(DIGITS, '0');
  document.getElementById("otp").innerText = otp;
  return otp;
}

// ================= XÁC THỰC OTP =================
async function verifyOTP() {
  const otp = await genOTP(); // Sinh lại OTP hiện tại
  result.innerText = (otp === userOtp.value)
    ? "OTP hợp lệ"
    : "OTP không hợp lệ";
}
</script>
```

</body>

</html>

# CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG THỬ NGHIỆM

## 4.1. Thiết kế và Hiện thực Giao diện (Front-end)

Dựa trên thuật toán TOTP đã được xây dựng bởi chương 3 (Sử dụng HMAC-SHA1 và Dynamic Truncation), nhóm tiến hành xây dựng giao diện demo dưới dạng ứng dụng Web để minh họa trực quan quá trình xác thực hai yếu tố.

### 4.1.1. Công nghệ sử dụng

- HTML5/CSS3: Xây dựng bố cục và giao diện người dùng, sử dụng Flexbox để phân chia khu vực tương tác.
- JavaScript (Client-side): Tích hợp module sinh OTP từ G3, xử lý logic đếm ngược thời gian thực (Real-time) và bắt sự kiện kiểm tra mã.
- Web Crypto API: Sử dụng thư viện bảo mật có sẵn của trình duyệt để thực hiện hàm băm HMAC-SHA1 tốc độ cao.

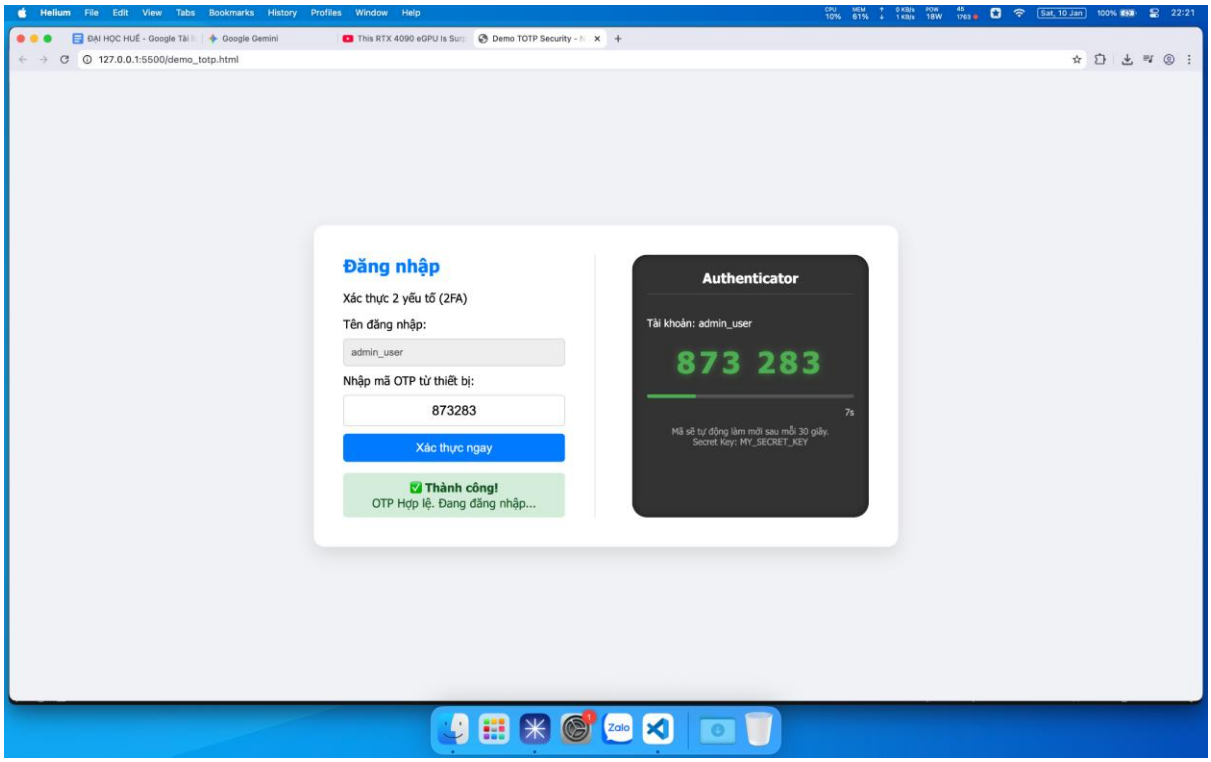
### 4.1.2. Mô tả các thành phần giao diện Ứng dụng demo được chia thành hai phân hệ hoạt động song song trên cùng một màn hình để thuận tiện cho việc kiểm thử:

1. Phân hệ Khách hàng (Client Login - Bên trái):
   - Mô phỏng giao diện đăng nhập của website ngân hàng hoặc mạng xã hội.
   - Bao gồm ô nhập liệu (Input field) chỉ chấp nhận 6 chữ số.
   - Nút "Xác thực" (Verify) gửi yêu cầu kiểm tra mã OTP tới hệ thống.
   - Khu vực hiển thị thông báo kết quả (Thành công/Thất bại) ngay lập tức.
2. Phân hệ Token ảo (Virtual Authenticator - Bên phải):
   - Mô phỏng một thiết bị phần cứng (Hard Token) hoặc ứng dụng điện thoại (như Google Authenticator).
   - Hiển thị OTP: Mã 6 số được sinh ra từ *Secret Key* và *Thời gian thực*.
   - Thanh tiến trình (Timer Bar): Trực quan hóa cửa sổ thời gian 30 giây (Time step). Thanh này sẽ tự động chạy lùi và đổi màu đỏ khi mã sắp hết hạn (còn dưới 5 giây).
   - Cơ chế tự động làm mới (Refresh) mã OTP khi hết chu kỳ 30 giây mà không cần người dùng tải lại trang.

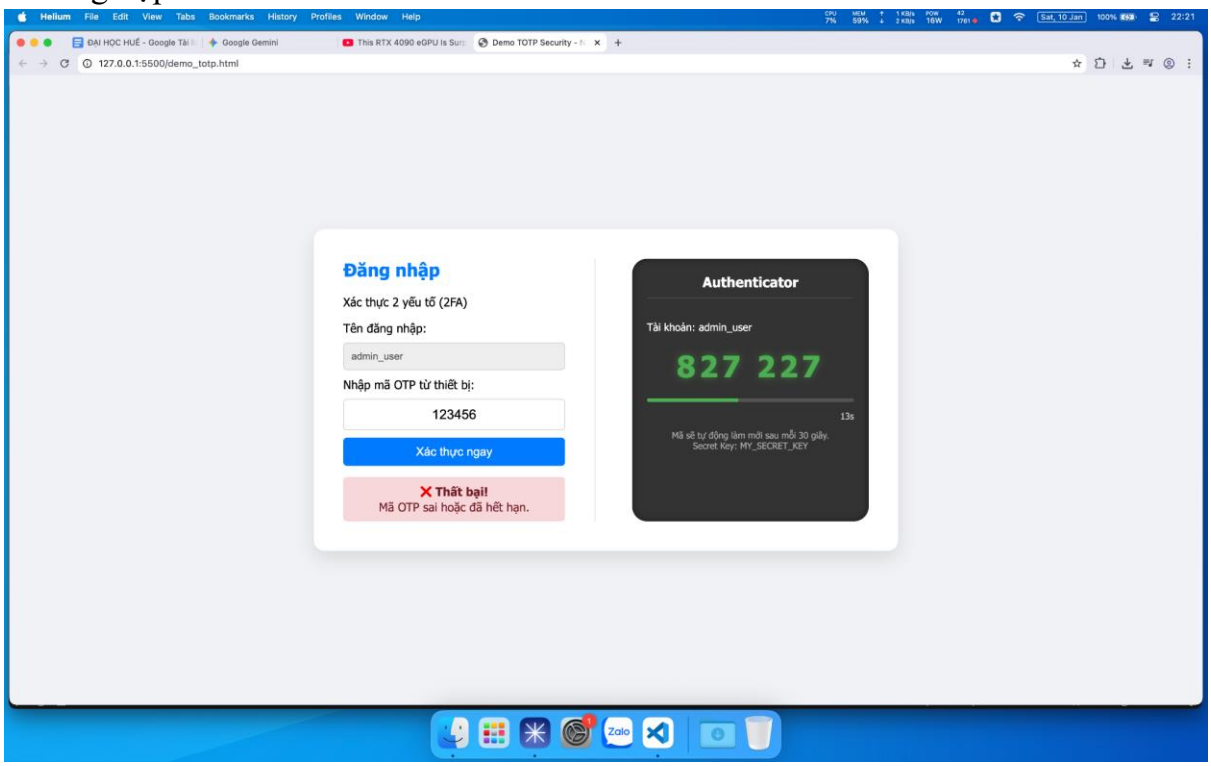
### 4.1.3. Kịch bản kiểm thử trên giao diện

- Trường hợp 1 (Hợp lệ): Người dùng nhìn mã bên phải (ví dụ: 123 456), nhập sang bên trái trong thời gian hiệu lực -> Hệ thống báo "Thành công".
- Trường hợp 2 (Sai mã): Người dùng nhập sai số -> Hệ thống báo "Mã OTP sai".
- Trường hợp 3 (Hết hạn): Người dùng nhìn mã (ví dụ: 123 456) nhưng đợi thanh thời gian chạy hết và đổi sang mã mới, sau đó mới bấm xác thực -> Hệ thống báo "Mã OTP sai hoặc đã hết hạn".

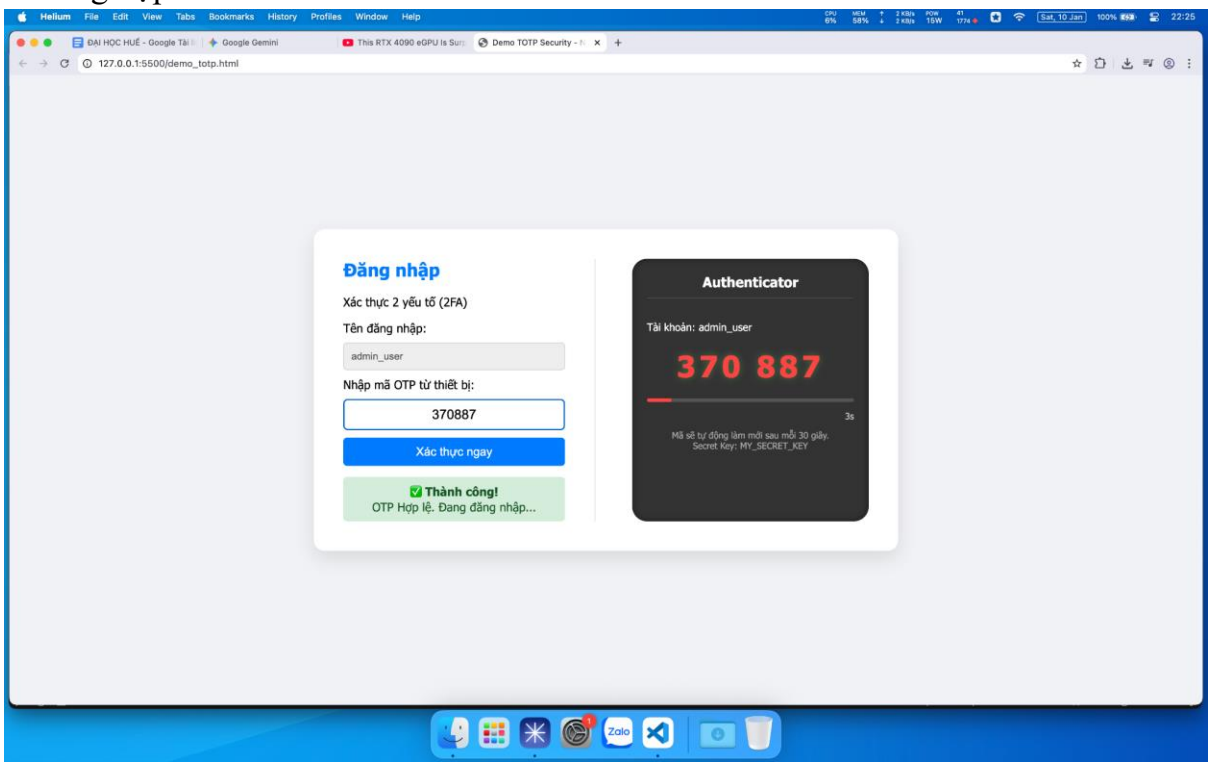
Trường hợp 1:

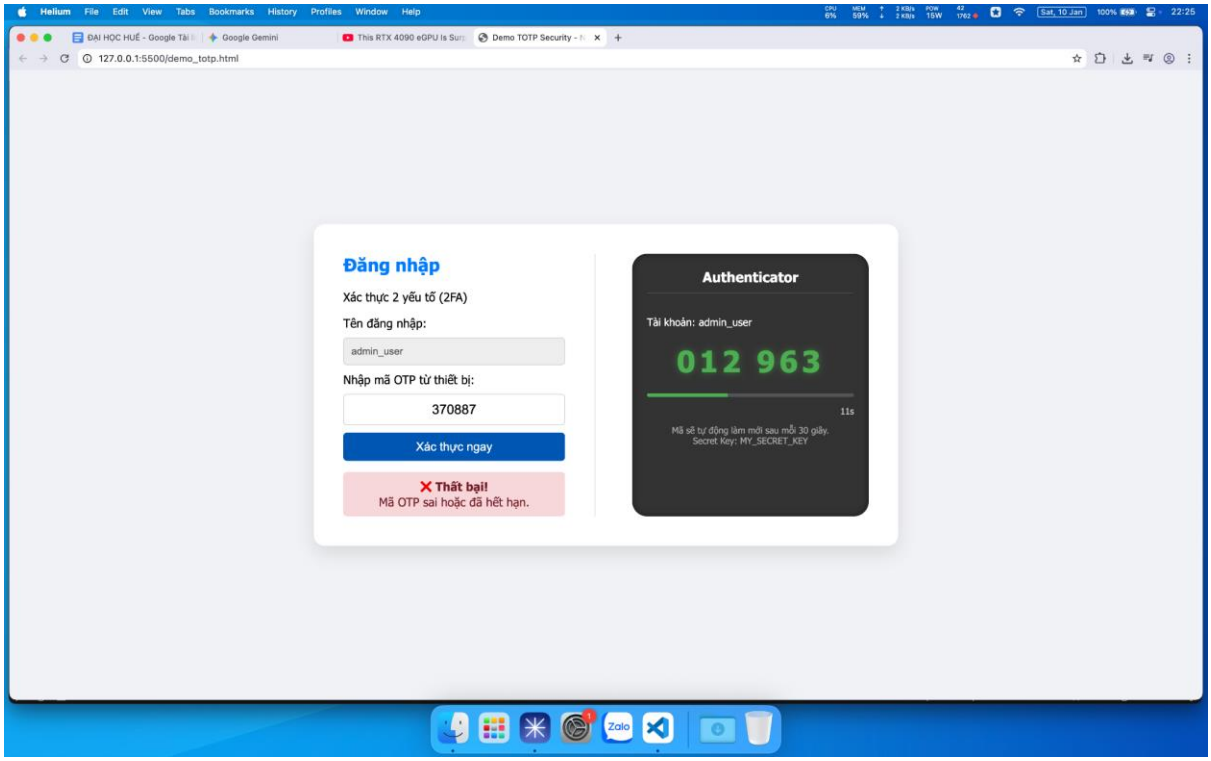


Trường hợp 2:




Trường hợp 3:






## Chương 5. Phân tích rủi ro bảo mật

5.1 Rủi ro lộ khóa bí mật (Secret Key)

Mô tả:

Khóa bí mật (Secret Key) là thành phần quan trọng nhất trong thuật toán TOTP, được sử dụng làm đầu vào cho hàm băm HMAC để sinh mã OTP. Trong hệ thống hiện tại, khóa bí mật có thể được lưu trữ dưới dạng văn bản rõ (plain text) trong mã nguồn hoặc cơ sở dữ liệu.

Nguy cơ:

Nếu khóa bí mật bị lộ do lỗi lập trình, rò rỉ mã nguồn hoặc bị truy cập trái phép vào hệ thống máy chủ, kẻ tấn công có thể tự sinh ra các mã OTP hợp lệ mà không cần sự cho phép của người dùng. Điều này làm mất hoàn toàn ý nghĩa của cơ chế xác thực hai yếu tố.Mức độ rủi ro: Cao

Biện pháp giảm thiểu:

- Mã hóa khóa bí mật khi lưu trữ.
- Không ghi khóa trực tiếp trong mã nguồn.
- Dùng xác thực 2 yếu tố (2FA)
- Sử dụng các cơ chế quản lý khóa an toàn (Key Management).
- Hạn chế quyền truy cập vào khóa bí mật trên máy chủ.

5.2 Rủi ro tấn công brute-force OTP (Brute Force Attack )

Mô tả:
Mã OTP trong hệ thống TOTP thường chỉ gồm 6 chữ số, tương ứng với khoảng 1.000.000 khả năng khác nhau. Do không gian tìm kiếm nhỏ, kẻ tấn công có thể thử nhiều giá trị OTP liên tiếp trong một khoảng thời gian ngắn.
Nguy cơ:
Nếu hệ thống không giới hạn số lần nhập OTP, kẻ tấn công có thể thực hiện tấn công brute-force để đoán đúng OTP trước khi mã hết hạn. Điều này làm tăng nguy cơ truy cập trái phép vào tài khoản người dùng.Mức độ rủi ro: Trung bình
Biện pháp giảm thiểu:

- Giới hạn số lần nhập OTP sai.
- Tạm khóa tài khoản khi nhập sai OTP quá số lần cho phép.
- Bắt buộc sử dụng mật khẩu mạnh
- Sử dụng CAPTCHA/reCAPTCHA
- Giám sát đăng nhập bất thường

5.3 Rủi ro lệch thời gian hệ thống (Systematic time-mismatch risk)
Mô tả:
Thuật toán TOTP sinh mã OTP dựa trên thời gian hệ thống. Nếu thời gian giữa máy chủ và thiết bị người dùng không được đồng bộ chính xác, mã OTP được sinh ra ở hai phía có thể khác nhau.
Nguy cơ:
Sự lệch thời gian có thể khiến OTP hợp lệ bị từ chối hoặc trong một số trường hợp hiếm, OTP cũ vẫn được chấp nhận. Điều này làm giảm tính sẵn sàng của hệ thống và gây khó khăn cho người dùng khi xác thực.Mức độ rủi ro: Thấp – Trung bình
Biện pháp giảm thiểu:

- Đồng bộ thời gian bằng NTP.
- Cho phép xác thực OTP lệch ±1 chu kỳ (±60 giây).
- Kiểm tra và hiệu chỉnh thời gian hệ thống định kỳ.

5.4 Rủi ro tấn công replay (Replay attack)
Mô tả: Là một cuộc tấn công mạng, trong đó kẻ tấn công sẽ nghe lén, ghi lại và sau đó phát lại một gói dữ liệu hợp lệ đã được truyền đi trước đó. Mục đích của chúng là để lừa hệ thống tin rằng đây là một giao tiếp hợp lệ từ người dùng, từ đó thực hiện các hành vi trái phép như chiếm quyền truy cập, chuyển tiền, hoặc thay đổi dữ liệu. Nguy cơ: Truy cập trái phép nếu OTP chưa hết hạn.
Mức độ rủi ro: Trung bình
Biện pháp giảm thiểu:

- Chỉ cho phép mỗi OTP được sử dụng một lần.
- Kết hợp OTP với phiên đăng nhập (session).

- Giám sát và phản ứng nhanh với các  bất thường

## Chương 6 :Thực nghiệm, đánh giá, hạn chế hướng đi và phát tiển

### 6.1 Thực nghiệm

Hệ thống TOTP được triển khai và chạy thử nghiệm trên máy tính cá nhân trong môi trường Java. Quá trình thực nghiệm tập trung vào việc đăng nhập và xác thực mã OTP trong nhiều trường hợp khác nhau, bao gồm OTP hợp lệ, OTP sai giá trị, OTP hết hạn , chứng tỏ tính hiệu quả và phù hợp của việc triển khai trên máy tính cá nhân.
Kết quả thực nghiệm cho thấy hệ thống sinh OTP hoạt động ổn định, các mã OTP có độ dài 6 chữ số và tự động thay đổi sau mỗi 30 giây, phù hợp với nguyên lý hoạt động của thuật toán TOTP theo chuẩn RFC 6238.Giúp tự động hóa quy trình , phát hiện lỗi sớm và đảm bảo chất lượng ứng dụng, làm nền tảng cho các hệ thống bảo mật có đọ tin cậy cao.

### 6.2 Đánh giá

Qua quá trình thực nghiệm, hệ thống cho thấy khả năng hoạt động ổn định và chính xác, đáp ứng tốt yêu cầu xác thực cơ bản bằng mã OTP. Các trường hợp OTP không hợp lệ như OTP sai giá trị, OTP hết hạn đều được hệ thống phát hiện và từ chối, qua đó nâng cao mức độ bảo mật so với phương pháp sử dụng mật khẩu tĩnh.
Hệ thống TOTP phù hợp cho mục đích học tập và nghiên cứu, giúp minh họa rõ nguyên lý hoạt động của cơ chế xác thực hai yếu tố dựa trên thời gian.

### 6.3 Hạn chế

Bên cạnh những kết quả đạt được, hệ thống vẫn còn một số hạn chế như:

- Khóa bí mật được sử dụng dưới dạng chuỗi đơn giản, chưa áp dụng chuẩn Base32.
- Chỉ kiểm tra OTP tại đúng một chu kỳ thời gian, chưa cho phép sai lệch thời gian.
- Chưa tích hợp các biện pháp bảo mật nâng cao như giới hạn số lần nhập OTP hoặc tạm khóa tài khoản.
- Thuật toán HMAC-SHA1 đã cũ, mức độ an toàn không còn cao trong các hệ thống hiện đại.

### 6.4 Hướng phát triển

Trong thời gian tới, hệ thống TOTP có thể được cải tiến và mở rộng theo một số hướng nhằm nâng cao tính bảo mật và khả năng ứng dụng thực tế. Trước hết, việc áp dụng chuẩn khóa bí mật Base32 sẽ giúp hệ thống tương thích với các ứng dụng xác thực phổ biến như Google Authenticator. Bên cạnh đó, cho phép xác thực OTP lệch ±30 giây sẽ giúp giảm ảnh hưởng của sự sai lệch thời gian giữa máy chủ và thiết bị người dùng.

Ngoài ra, hệ thống cần bổ sung cơ chế giới hạn số lần nhập sai OTP và tạm khóa tài khoản khi vượt quá ngưỡng cho phép để hạn chế các cuộc tấn công brute-force. Việc nâng cấp thuật toán băm lên HMAC-SHA256 hoặc HMAC-SHA512 cũng là hướng đi cần thiết nhằm đáp ứng yêu cầu bảo mật của các hệ thống hiện đại. Cuối cùng, hệ thống TOTP có thể được tích hợp vào các ứng dụng xác thực hai yếu tố trong thực tế để nâng cao mức độ an toàn cho người dùng.

6.4.1. Áp dụng chuẩn khóa bí mật Base32

Để đảm bảo khả năng tương thích với Google Authuenticator, hệ thống cần chuyển đổi các khóa bí mật (shared secrets) sang định dạng Base32.

- Đặc điểm: Base32 loại bỏ các ký tự dễ gây nhầm lẫn như 0, 1, I, O, giúp người dùng dễ dàng nhập liệu thủ công setup case khi không thể quét mã QR.
- Tiêu chuẩn: Tuân thủ RFC 4648 để đảm bảo mã QR sinh ra có thể được nhận diện chính xác bởi mọi ứng dụng TOTP phổ biến hiện nay.

6.4.2. Xác thực OTP lệch ±30 giây

Cải tiến này giải quyết vấn đề lệch thời gian (Time Drift) giữa thiết bị người dùng và máy chủ, vốn là nguyên nhân phổ biến gây lỗi "Mã không hợp lệ".

- Cơ chế: Hệ thống sẽ kiểm tra mã OTP của chu kỳ hiện tại cùng với mã của chu kỳ ngay trước và ngay sau đó (tổng cửa sổ thời gian là 90 giây).
- Lợi ích: Giảm thiểu tỷ lệ xác thực thất bại khi đồng hồ điện thoại người dùng chạy nhanh hoặc chậm hơn thực tế mà vẫn đảm bảo tính an toàn.

6.4.3. Giới hạn số lần nhập sai và tạm khóa tài khoản

Để ngăn chặn các cuộc tấn công Brute-force (thử sai liên tục), hệ thống cần thiết lập ngưỡng an toàn.

- Quy trình: Tài khoản sẽ bị khóa tạm thời sau 5 lần nhập sai liên tiếp.
- Xử lý: Người dùng phải chờ một khoảng thời gian nhất định (ví dụ 15-30 phút) hoặc liên hệ hỗ trợ để mở khóa, giúp bảo vệ tài khoản khỏi các truy cập trái phép.

6.4.4. Nâng cấp thuật toán băm lên HMAC-SHA256/512

Thay vì sử dụng SHA-1 (vốn đã cũ và tiềm ẩn rủi ro), việc nâng cấp lên các thuật toán mạnh hơn là xu hướng bắt buộc trong năm 2026.

- HMAC-SHA256: Cung cấp độ bảo mật cao hơn với giá trị băm 256-bit, được thiết kế bởi NSA và là tiêu chuẩn an toàn hiện nay.
- HMAC-SHA512: Tối ưu cho các hệ thống yêu cầu mức độ bảo mật khắt khe nhất và hiệu suất tốt trên các kiến trúc phần cứng 64-bit.
- Tiêu chuẩn áp dụng: Tuân thủ danh mục tiêu chuẩn kỹ thuật mật mã hiện hành như TCVN 11495-2 để đảm bảo tính pháp lý và an toàn dữ liệu.

TÀI LIỆU THAM KHẢO

[1] https://viettelidc.com.vn/tin-tuc/otp-la-gi, truy cập ngày 08/01/2026.

[2] https://timo.vn/blogs/timo-debit-atm-napas/ma-otp-sms-la-gi-cach-lay-ma-otp/

[3] https://www.thegioididong.com/hoi-dap/ma-otp-la-gi-co-may-loai-dung-de-lam-gi-4-luu-y-khi-su-1309796#hmenuid2 ("Mã OTP là gì? Các loại mã OTP & Cách lấy mã OTP trên điện thoại")

[4] https://stringeex.com/vi/blog/post/sms-otp-la-gi

[5] https://indochinatelecom.vn/tim-hieu-voice-otp/

[6] https://www.vietguys.biz/vi/martech/knowledge/email-otp-la-gi-giai-phap-xac-thuc-nhanh-bao-mat-an-toan-cho-moi-doanh-nghiep

[7] https://hdbank.com.vn/vi/news/detail/tin-tuc-khac/token-la-gi

[9] https://the-simple.jp/en-what-is-hotp-hmac-based-one-time-password-explanation-of-one-time-password-technology

[10] https://www.sharetru.com/blog/one-time-password-otp-authentication-methods-you-should-know-hotp-totp

[11] https://fptshop.com.vn/tin-tuc/danh-gia/totp-la-gi-185553

[12] https://www.1kosmos.com/security-glossary/time-based-one-time-password-totp/