ISSN 2354-0575

NGHIÊN CỨU VÀ XÂY DỰNG HỆ THỐNG GIÁM SÁT WEBSITE TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

Vũ Xuân Thắng, Đặng Vân Anh, Trần Đỗ Thu Hà
Trường Đại học Sư phạm Kỹ thuật Hưng Yên

Ngày tòa soạn nhận được bài báo: 10/10/2017
Ngày phản biện đánh giá và sửa chữa: 12/11/2017
Ngày bài báo được chấp nhận đăng: 15/11/2017

Tóm tắt:

Trong bài báo chúng tôi trình bày quá trình thiết kế, cài đặt và thử nghiệm một giải pháp bảo đảm an toàn cho website của trường ĐH SPKT Hưng Yên. Cụ thể chúng tôi nghiên cứu các phương pháp tấn công và các giải pháp phòng chống tấn công website trên Internet. Nhóm nghiên cứu đã đề xuất giải pháp chống tấn công thay đổi nội dung từ việc cải tiến thuật toán Rabin Fingerprint áp dụng cho các website trường ĐH SPKT Hưng Yên, từ đó có thể mở rộng cho các website khác trên internet.

Từ khoá: *web security, Rabin fingerprint application, web protected.*

1. Giới thiệu chung

Ngày nay nhu cầu truyền tải thông tin lớn, việc ứng dụng các website, các phương tiện quảng bá thông tin không chỉ dừng lại ở doanh nghiệp mà còn hướng đến cả người dùng cá nhân. Từ đó, nhu cầu đảm bảo an toàn thông tin truyền tải trên các website trở thành chủ đề nóng trong xã hội. Nắm bắt xu thế đó, nhóm tác giả đã thực hiện nghiên cứu các vấn đề về tấn công và cách thức phòng tránh các cuộc tấn công vào các website trên internet.

Để tấn công vào các website hacker đã thực hiện khai thác một số lỗ hổng từ Hệ điều hành, hệ quản trị cơ sở dữ liệu, các dịch vụ internet, lỗi lập trình… [1]

Các lỗ hổng bảo mật trên một hệ thống là các điểm yếu có thể tạo ra sự ngưng trệ của dịch vụ, thêm quyền đối với người sử dụng hoặc cho phép các truy nhập không hợp pháp vào hệ thống. Các lỗ hổng cũng có thể nằm ngay các dịch vụ như Sendmail, Web, Ftp... Ngoài ra các lỗ hổng còn tồn tại ngay chính tại hệ điều hành như trong Windows, Unix hoặc trong các ứng dụng mà sử dụng thường xuyên như Word, Excel,... [2]

Trong phần tiếp theo chúng tôi tổng kết những kiểu tấn công phổ biến nhằm vào các website. Chúng tôi cũng nhắc lại cách tiếp cận giải pháp chủ yếu được áp dụng để phòng, chống những kiểu tấn công trên. Đây là cơ sở hình thày và phát triển giải pháp giám sát website và những ý tưởng chính của nhóm tác giả. Quá trình thiết kế, thực nghiệm đã phản ánh được kết quả khả quan của giải pháp đưa ra trong bài báo.

2. Kỹ thuật tấn công Website và giải pháp phòng chống

2.1. Tấn công SQL Injection

SQL Injection là một kĩ thuật cho phép những kẻ tấn công lợi dụng lỗ hổng trong việc kiểm tra dữ liệu nhập trong các ứng dụng web, các thông báo lỗi của hệ quản trị cơ sở dữ liệu để "tiêm vào" (inject) và thi hành các câu lệnh SQL bất hợp pháp gây ảnh hưởng tới dữ liệu người dùng. Lỗi này thường xảy ra trên các ứng dụng Web có dữ liệu được quản lí bằng các hệ quản trị cơ sở dữ liệu như SQL Server, MySQL, Oracle,... [3]

Có bốn dạng thông thường bao gồm: Tấn công xác thực (authorization bypass), sử dụng câu lệnh SELECT, sử dụng câu lệnh INSERT, sử dụng các stored-procedures [2].

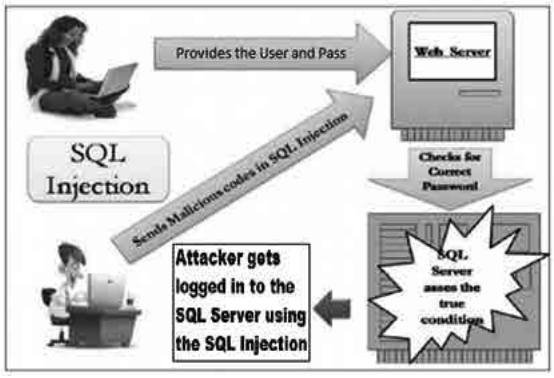
- Tấn công xác thực (authorization bypass): với dạng tấn công này, tin tặc có thể dễ dàng vượt qua các trang đăng nhập nhờ vào lỗi khi dùng các câu lệnh SQL trên cơ sở dữ liệu ứng dụng web.

- Dạng tấn công sử dụng câu lệnh SELECT: Dạng tấn công này kẻ tấn công lợi dụng các sơ hở trong các thông báo lỗi từ hệ thống để dò tìm các điểm yếu từ đó thực hiện tấn công qua câu truy vấn dữ liệu.

- Dạng sử dụng câu lệnh INSERT: thông thường các ứng dụng web cho phép người dùng đăng kí một tài khoản để tham gia. Chức năng không thể thiếu là sau khi đăng kí thành công, người dùng có thể xem và hiệu chỉnh thông tin của mình. SQL Injection có thể được dùng khi hệ thống không kiểm tra tính hợp lệ của thông tin nhập vào.

- Dạng tấn công sử dụng stored-procedures:

việc tấn công bằng stored-procedures sẽ gây tác hại rất lớn nếu ứng dụng được thực thi với quyền quản trị hệ thống 'sa'. Ví dụ, nếu ta thay đoạn mã tiêm vào dạng: '; EXEC xp_cmdshellcmd.exe dir C: '. Lúc này hệ thống sẽ thực hiện lệnh liệt kê thư mục trên ổ đĩa C:\ cài đặt server. Việc phá hoại kiểu nào tuỳ thuộc vào câu lệnh đằng sau cmd.exe.



Hình 2.1. *Mô hình tấn công SQL Injection*

Để phòng tránh các lỗi SQL injection ta có thể thực hiện ở hai mức:

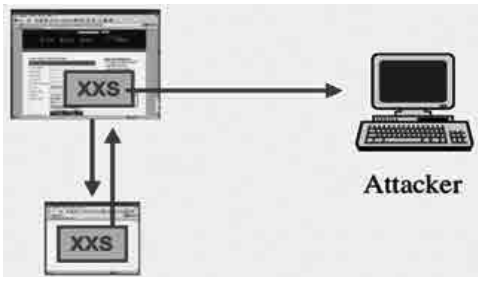
- Kiểm soát chặt chẽ dữ liệu nhập vào: Để phòng tránh các nguy cơ có thể xảy ra, hãy bảo vệ các câu lệnh SQL là bằng cách kiểm soát chặt chẽ tất cả các dữ liệu nhập nhận được từ đối tượng Request.

- Thiết lập cấu hình an toàn cho hệ quản trị cơ sở dữ liệu.

2.2. Tấn công XSS (Cross Site Scripting)

Cross-Site Scripting (XSS) là một trong những kĩ thuật tấn công phổ biến nhất hiên nay, đồng thời nó cũng là một trong những vấn đề bảo mật quan trọng đối với các nhà phát triển web và cả những người sử dụng web. Ngay cả đối với những trang như www.fbi.gov, www.yahoo.com... cũng đã từng bị lỗi XSS. Bất kì một website nào cho phép người sử dụng đăng thông tin mà không có sự kiểm tra chặt chẽ các đoạn mã nguy hiểm thì đều có thể tiềm ẩn các lỗi XSS [2]

Thường thì XSS có thể xảy ra ở chỗ nào mà người dùng có thể nhập dữ liệu vào và sau đó sẽ nhận được một thông báo trả về. Nên thường chúng ta sẽ kiểm tra ở những ô đăng nhập (login) đầu vào. Khi nhập một chuỗi kí tự nào đó mà kết quả của Server trả về có liên quan tới chuỗi mà bạn nhập thì rất có khả năng trang đó bị mắc lỗi XSS [5].



Hình 2.2. *Mô hình tấn công XSS*

Thông thường kẻ tấn công (attacker) sử dụng XSS để lấy các thông tin quan trọng: cookie, username, password. Ở đây sử dụng XSS để đánh cắp cookies của nạn nhân (victim).

- Cách chèn script:

+ Sử dụng Java script:

```
javascript: alert(document.cookie)
```

+ Dùng file.php:

```
javascript:location="http://hostcuaban/
cookie.php?cookie="+(docum ent.cookie)
```

Hàm location để chuyển trình duyệt đến 1 trang khác, lúc đó document.cookie sẽ thay bằng giá trị cookie.

Đối với ứng dụng web mã nguồn mở, bạn có thể tham khảo danh sách các lỗ hổng của chương trình của bạn trên các trang web chứa các thông tin về bảo mật như securityfocus.com, securiteam.com... Tuy nhiên nếu các website được tự viết mã nguồn thì bạn không thể áp dụng phương pháp trên. Trong trường hợp này bạn cần đến các chương trình dò tìm kiếm (scanner) tự động như: N-Stealth hay AppScan, đây là những chương trình quét tìm khá hiệu quả, bạn không chỉ kiểm tra được các lỗi XSS mà nó còn cho phép bạn kiểm tra các lỗi khác trong Website đó, server đó [5].

Có rất nhiều cách để có thể giải quyết vấn đề này như:

- Chỉ chấp nhận những dữ liệu hợp lệ.
- Từ chối nhận các dữ liệu hỏng.
- Liên tục kiểm tra và lọc dữ liệu.

2.3. Tấn công từ chối dịch vụ DOS

Tấn công DOS là một kiểu tấn công mà một người làm cho một hệ thống không thể sử dụng, hoặc làm cho hệ thống đó chậm đi một cách đáng kể với người dùng bình thường, bằng cách làm quá tải tài nguyên của hệ thống

Các kiểu tấn công thuộc phương thức này rất đa dạng [9]:

- Tấn công chiếm dụng tài nguyên (Resource Depletion): Bằng cách lạm dụng quá trình giao tiếp của giao thức mạng hoặc những gói tin dị thường, kẻ tấn công sẽ chiếm dụng nguồn tài nguyên hệ thống như bộ nhớ (RAM) và bộ vi xử lý (CPU)… khiến cho người dùng chia sẻ không truy xuất được hệ thống do hệ thống không đủ khả năng xử lý.

- Tấn công SYN: Được xem là một trong những kiểu tấn công DOS kinh điển nhất. Lợi dụng sơ hở của thủ tục TCP khi "bắt tay ba chiều", mỗi khi máy khách (client) muốn thực hiện kết nối (connection) với máy chủ (server) thì nó thực hiện việc bắt tay ba lần (three-ways-handshake) thông qua các gói tin (packet).

- Tấn công chiếm dụng băng thông (Bandwidth Depletion): Có 2 loại tấn công chính:

+ Thứ nhất là làm ngập bằng cách gửi liên tục các gói tin có kích thước lớn đến hệ thống nạn nhân, làm nghẽn băng thông nạn nhân.

+ Thứ hai là sử dụng mạng khuếch đại, kẻ tấn công gửi thông tin đến một địa chỉ phát quảng bá (broadcast), tất cả hệ thống trong mạng con nạn nhân (victim) trong phạm vi truy xuất địa chỉ quảng bá sẽ gửi lại hệ thống nạn nhân một thông điệp phản hồi cho máy nạn nhân chấp nhận trao đổi dữ liệu. Phương pháp này khuếch đại dòng lưu lượng làm cho hệ thống nạn nhân giảm băng thông đáng kể.

Có ba giai đoạn chính trong quá trình phòng chống tấn công DOS:

- Giai đoạn ngăn ngừa: tối thiểu hóa lượng Agent, tìm và vô hiệu hóa các Handler.

- Giai đoạn đối đầu với cuộc tấn công: Phát hiện và ngăn chặn cuộc tấn công, làm suy giảm và dừng cuộc tấn công, chuyển hướng cuộc tấn công.

- Giai đoạn sau khi cuộc tấn công xảy ra: thu thập chứng cứ và rút kinh nghiệm.

Các giai đoạn chi tiết trong phòng chống DOS:

- Tối thiểu hóa số lượng Agent: Từ phía người dùng (user), một phương pháp rất tốt để ngăn ngừa tấn công DOS là từng người dùng mạng sẽ tự đề phòng không để bị lợi dụng tấn công hệ thống khác. Muốn đạt được điều này thì ý thức và kỹ thuật phòng chống phải được phổ biến rộng rãi cho các người dùng mạng. Các người dùng mạng phải liên tục thực hiện các quá trình bảo mật trên máy vi tính của mình. Một số giải pháp tích hợp sẵn khả năng ngăn ngừa việc cài đặt đoạn mã (code) nguy hiểm vào phần cứng (hardware) và phần mềm (software) của từng hệ thống. Về phía người dùng mạng họ nên cài đặt và cập nhật liên tục các phần mềm phòng chống virus, các bản sửa lỗi của hệ điều hành [6].

- Tìm và vô hiệu hóa các bộ xử lý (handler): Một nhân tố vô cùng quan trọng trong tấn công mạng (attack-network) là bộ xử lý (handler), nếu có thể phát hiện và vô hiệu hóa bộ xử lý thì khả năng phòng chống tấn công DOS thành công là rất cao.

3. Xây dựng hệ thống đảm bảo an ninh Website cho trường ĐHSPKT Hưng Yên

3.1. Yêu cầu

Thực hiện 5 bước cơ bản cần thiết để duy trì an ninh cho Hệ điều hành (HĐH):

● Lập kế hoạch cài đặt, triển khai HĐH máy chủ và các thành phần khác cho Webserver đó.

● Vá và cập nhật HĐH máy chủ theo yêu cầu.

● Hardening (cứng hóa) và cấu hình HĐH máy chủ để giải quyết tương xứng vấn đề an ninh.

● Cài đặt và cấu hình kiểm soát bảo mật bổ sung (additional security controls) nếu cần thiết.

● Kiểm tra HĐH máy chủ để đảm bảo rằng bốn bước trước đó giải quyết đầy đủ tất cả các vấn đề an ninh.

3.2. Xây dựng chương trình đảm bảo an ninh hệ thống

Một trong những kiểu tấn công được biết rộng rãi nhất là tấn công thay đổi Website.Nó thường là các mã độc (virus, worm, trojan, và các loại mã độc khác), được thiết kế để xóa bỏ, sửa đổi, hoặc thay thế các trang web trên webserver.

Những cuộc tấn công thay đổi Website đã được thực hiện để xâm phạm tính toàn vẹn của Web bằng một trong những hình thức sau:

- Thay đổi nội dung của trang Web.
- Thay đổi một phần nội dung trang Web.
- Thay thế toàn bộ trang Web.
- Sử dụng lại trang Web cũ.
- Thay đổi bề ngoài của trang Web.
- Chuyển hướng trang web.
- Phá hủy hoặc xóa bỏ trang Web.

Bài báo đề xuất xây dựng hệ thống giám sát website nhằm phát hiện kịp thời các cuộc tấn công (ở trên) bằng hệ thống đa kiểm tra dựa trên thuật toán dấu vân tay nhanh (fast fingerprint algorithm) để đảm bảo tính toàn vẹn của trang web đồng thời tạo ra thông điệp cảnh báo có ý nghĩa và phục hồi lại các trang web đã bị tấn công.

3.3. Hàm băm mật mã

Hàm băm là nền tảng cho nhiều ứng dụng mã hóa. Có nhiều thuật toán để thực hiện hàm băm, trong số đó, phương pháp SHA-1 và MD5 thường được sử dụng khá phổ biến từ thập niên 1990 đến nay [4].

Hàm băm mật mã phải có khả năng chống lại các loại tấn công mật mã, tối thiểu phải đảm bảo có 3 tính chất sau:

● Kháng tiền ảnh (Pre-image resistance): Với một mã băm h bất kỳ, khó tìm được một thông điệp m nào mà h = hash(m).

● Kháng tiền ảnh thứ hai (Second pre-image resistance): Với một thông điệp m_1 bất kỳ, khó tìm được một thông điệp thứ hai m_2 sao cho $m_1 \neq m_2$ và $hash(m_1) = hash(m_2)$.

● Kháng xung đột (Collision resistance): Khó tìm được một cặp thông điệp m_1 và m_2 sao cho $m_1 \neq m_2$ và $hash(m_1) = hash(m_2)$.

Thực hiện:

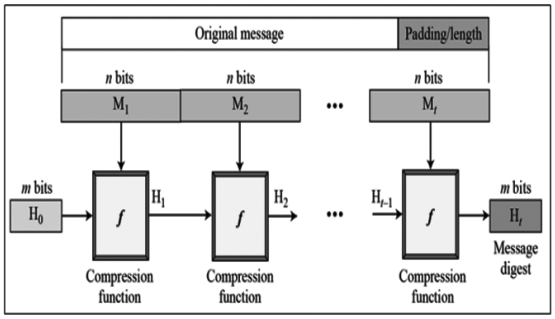
Bước 1: Gọi H là trạng thái có kích thước n bit, f là "hàm nén" thực hiện thao tác trộn khối dữ liệu với trạng thái hiện hành.

Bước 2: Khởi gán H_0 bằng một vector khởi

tạo nào đó.

Bước 3: $H_i = f(H_{i-1}, M_i)$ với i = 1, 2, 3, …, s

Khi đó: Hs chính là thông điệp rút gọn của thông điệp M ban đầu.



Hình 3.1. *Sơ đồ Merkel-Damgard*

3.4. Thuật toán Rabin Fingerprint

Thuật toán Rabin Fingerprint là một trong nhiều thuật toán Fingerprint thực hiện khóa công khai sử dụng các đa thức trên một trường giới hạn [10].

Thuật toán được sử dụng trong hệ thống như sau:

- Đầu vào: Tài liệu (trang web công khai)
- Đầu ra: Dấu vân tay tài liệu (các giá trị băm của tài liệu đó)

Bước 1: Bắt đầu.

Bước 2: Xử lý văn bản, xoá hết tất cả khoảng trắng và các kí tự đặc biệt (như: <,>, %,!, …).

Bước 3: Chia khối văn bản đã xử lý đó thành các chuỗi con có độ dài K.

// Số lượng chuỗi con có độ dài K và số lượng giá trị băm (mã băm) bằng (m-K+1), với m là kích thước của tài liệu.

Bước 4: Tính toán giá trị băm đối với mỗi chuỗi con bằng cách tính H(P) như sau:

// H(P) là một tuyến tính trong n (n là độ dài của P)

Bước 5: Lưu lại tất cả các giá trị băm của văn bản.

Bước 6: Kết thúc.

3.5. Thuật toán Rabin Fingerprint cải tiến

Thuật toán cải tiến được đề xuất trong hệ thống như sau:

Đầu vào: Tài liệu (trang web công khai)

Đầu ra: Dấu vân tay tài liệu (các giá trị băm của tài liệu đó)

Bước 1: Bắt đầu.

Bước 2: Xử lý văn bản, xoá hết tất cả khoảng trắng và các kí tự đặc biệt (như: <, >, %, !, …) từ mã HTML (mã trang web) để thu được một khối văn bản thuần túy (pure text block).

Bước 3: Chia văn bản M thành K khối, mỗi khối con có kích thước là n. K=m/n với m là kích thước của văn bản M, n là số nguyên dương cho trước là kích thước của mỗi chuỗi con.

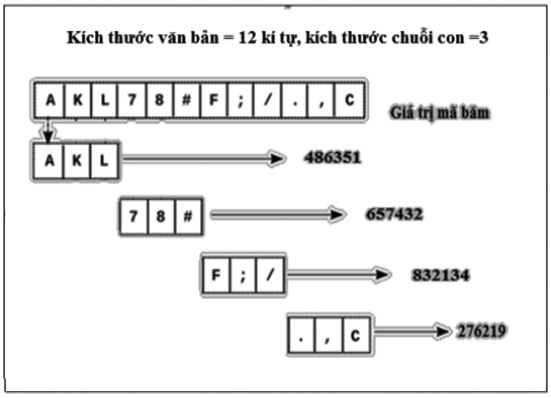
Bước 4: Tính mã băm H(P) cho các chuỗi con như sau:

Khởi tạo:

```
Tr = T[r..r+n-1];
K=0;
H(S) = S(n) + 2*S(n-1) + 4*S(n-2) + … + (2n-1)*S(1);
While (K<m/n)
{
        for (r=K*n; r<=(K*n+n); r++){
        Hp(Tr)= (Hp(Tr) + T(r)) mod p
        //Tính gt băm cho các chuỗi con, p là nt lớn.
}
K++;
}
```

Bước 5: Lưu lại tất cả các giá trị băm của văn bản.

Bước 6: Kết thúc.



Hình 3.3. *Minh hoạ cải tiến giải thuật*

3.6. Hệ thống giám sát Website

Mục đích chính của hệ thống giám sát website (Anti Website Defacement System) là để phát hiện bất kỳ các cuộc tấn công thay đổi web nào và phục hồi các tập tin của web đã bị tấn công. Để đạt được nhiệm vụ này, hệ thống giám sát website được thiết kế và triển khai trên 2 máy chủ (Web-server và AWDS-server) với 5 hệ thống con (subsystem) được tích hợp và cơ sở dữ liệu tập trung.

Các bản sao mới nhất của các trang web được lưu trữ trong một khu vực bên ngoài máy chủ Web (Web-server), trên một máy chủ khác đặt tại một vùng mạng khác. Các thành phần đó và mối quan hệ giữ chúng được thể hiện trong sơ đồ sau:

Hệ thống Builder hoạt động tại AWDS-server khi hệ thống Admin thêm vào, chỉnh sửa, hoặc loại bỏ các trang web. Hệ thống Builder hoạt

động như sau:

- Tính toán giá trị băm và các thuộc tính (tên, kích thước của tập tin, loại, ngày chỉnh sửa) của các trang web đã cho (trang web mới được thêm vào hoặc trang web được chỉnh sửa).

- Lưu những thông tin tính toán vào trong danh sách lưu trữ CSDL, và hệ thống cung cấp trang web (hệ thống Builder) có nhiệm vụ duy trì bảo quản danh sách này.

- Lưu lại các trang web này trong thư mục phục hồi tại máy chủ AWDS để phục hồi lại trang web như ban đầu nếu phát hiện có sự tấn công làm thay đổi về nội dung.

- Công bố các trang web đã cho ở thư mục Input tại máy chủ web.

Hệ thống con Multi-checker (đa kiểm tra) là công việc chính của hệ thống giám sát website, nó chứa vài kiểm tra làm việc bên cạnh thư mục Input tại Web-server.

Hệ thống sẽ thường xuyên kiểm tra định kỳ tính toàn vẹn của các trang web được công bố, và các trang web quan trọng như trang chủ (index. html, default.asp, default.aspx…). Do đó có một khoảng thời gian liên quan tới mỗi tập tin để kiểm tra lại tính toàn vẹn [8].

Hệ thống hoạt động phụ thuộc vào một trong hai chế độ sau:

- Chế độ kiểm tra cơ bản (basic mode): Khi hệ thống đa kiểm tra bắt đầu hoạt động, hệ thống sẽ kiểm tra các thuộc tính (tên, kích thước của tập tin, loại, và ngày chỉnh sửa) cho trang web hiện tại (trang web được lưu trong Input) và so sánh nó với trang web đã được lưu trữ trước của chính nó trong CSDL.

- Chế độ kiểm tra nâng cao (advanced mode): Khi hệ thống Multi-checker bắt đầu hoạt động, hệ thống sẽ tính toán tìm giá trị băm (sử dụng thuật toán cải tiến Rabin Fingerprint) cho các trang web được công bố (được lưu trữ trên thư mục Intpub) và so sánh nó với trang web đã được lưu trữ trước của chính nó trong CSDL.

4. Cài đặt và thử nghiệm chương trình

4.1. Yêu cầu

Chương trình thử nghiệm được phát triển bằng ngôn ngữ PHP và hệ quản trị CSDL MariaDB. Với cấu hình máy sử dụng là:

- Bộ xử lý: Intel(R) Core(TM)2 Duo CPU T6670 @ 2.20GHz
- Bộ nhớ Ram: 4.00 GB.
- Hệ điều hành: Ubuntu 16.04

4.2. Thực nghiệm

Chương trình được thử nghiệm kiểm tra về thời gian xử lý của thuật toán Rabin Fingerprint và thuật toán cải tiến Rabin Fingerprint với dữ liệu vào là 4 website (sử dụng hàm stopwatch() trong C# để đo thời gian xử lý của thuật toán).

Kết quả thử nghiệm của chương trình với 3 trang web về thời gian tính toán của thuật toán Rabin Fingerprint và cải tiến thuật toán Rabin Fingerprint như bảng biểu sau:

Bảng 1. *Kết quả kiểm tra bằng Rabit Fingerprint*

Website	Rabin fingerprint (Thời gian)	Cải tiến Rabin fingerprint (Thời gian)	Kích thước
utehy.edu.vn	00:07:52.26 59048	00:00:00.017 9612	197342
fit.utehy.edu. vn	00:10:56.65 08695	00:00:00.023 7493	260318
tuyensinh. utehy.edu.vn	00:00:29.77 48576	00:00:00.005 3756	34798

Chương trình được thử nghiệm kiểm tra về thời gian xử lý của hệ thống kiểm tra cơ bản và kiểm tra chi tiết với dữ liệu vào là 3 trang web.

Kết quả thử nghiệm của hệ thống kiểm tra cơ bản và kiểm tra nâng cao như bảng sau:

Bảng 2. *Kết quả kiểm tra của hệ thống mới*

Website	Basic mode (Thời gian)	Advanced mode (Thời gian)	Kích thước
utehy.edu.vn	00:00:00.00 13731	00:00:00.0038 273	197342
fit.utehy.edu. vn	00:00:00.00 21435	00:00:00.0042 867	260318
tuyensinh. utehy.edu.vn	00:00:00.00 10857	00:00:00.0026 319	96819

Chương trình được thử nghiệm theo dõi giám sát đa luồng, cùng một lúc theo dõi 4 website. Chức năng Multi-thread đã hoạt động tốt, đưa ra được cảnh báo khi có sự thay đổi nội dung trang web.

Chương trình được thử nghiệm kiểm tra ở chế độ nâng cao, đã chỉ ra được vị trí thay đổi của nội dung trang web.

4.3. Nhận xét kết quả

- Thời gian xử lý của thuật toán cải tiến Rabin Fingerprint là nhanh hơn rất nhiều so với thuật toán Rabin Fingerprint (đặc biệt khi kích thước trang web lớn).

- Thời gian xử lý của hệ thống Multi-checker ở hai chế độ kiểm tra cơ bản (basic mode) và kiểm tra nâng cao (advanced mode) là tương đương nhau vì cùng độ phức tạp thuật toán.

- Chương trình đã giám sát được sự thay đổi bất thường của website khi theo dõi đơn tiến trình (một website) cũng như đa tiến trình (nhiều website

đồng thời), và đã đưa ra được cảnh báo hợp lý.

- Ở chế độ Advanced Mode đã chỉ ra được vị trí thay đổi của nội dung trang web.

5. Kết luận

Bài báo đã giới thiệu tổng quan về Webserver và Website, đồng thời phân tích các lỗ hổng an ninh trên Web dẫn tới những kiểu tấn công Web phổ biến trên thế giới, và cũng trình bày các kỹ thuật phòng chống các kiểu tấn công đó.

Phân tích sâu một số phương pháp đảm bảo an ninh Web: Đảm bảo an ninh HĐH webserver, đảm bảo an ninh webserver, đảm bảo an ninh nội dung web, sử dụng kỹ thuật xác thực và mã hóa, triển khai cơ sở hạ tầng mạng an ninh, quản trị webserver.

Bài báo đã đề xuất, xây dựng được hệ thống giám sát website có thể theo dõi đa luồng (nhiều website đồng thời) và có thể giám sát được những trang web động, đưa ra cảnh báo kịp thời có ý nghĩa.

Tài liệu tham khảo

[1]. Hans Delfs and Helmut Knebl, "*Introduction to Cryptoraphy*", 2nd Edition, Springer – 2007.
[2]. Miles Tracy, Wayne Jansen, Karen Scarfone, Theodore Winograd, "*Guidelines on Securing Public Web Servers*", Version 2, NIST – September 2007.
[3]. Charles P. Pfleeger and Shari Lawrence, "*Security in Computing*", 3rd Edition, Prentice Hall– 2003.
[4]. William Stalling, "*Cryptography and Network Security*", Prentice Hall – 1999.
[5]. Amanda Andress and Mandy Andress, Sams, "*Surviving Security: How to Integrate People*", process, and technology, 2nd Edition, 2004.
[6]. E.L.Cashin, "*Integerit file Verification System*", 2000.
[7]. Rocksoft, "*Veracity- nothing can change without you knowing: Data Integrity Assurance*", 2003.
[8]. Selvitri F, "*High Performance Issues in Web Search Engines*", 2004.
[9]. V.A.Narayana, P.Premchnd, IEEE International Advance Computing Conference (IACC2009), Patiala, India, "*A Novel and Efficient Approach for Near Duplicate Page Detection in Web Crawling*", 6-7 March 2009.
[10]. A. Z. Broder, Springer-Verlag, "*Some Applications of Rabin's Fingerprinting Method*", 1993.
[11]. Rabin-Karp Algorithm, "*Rolling Hash*", February 18, 2011.
[12]. Mark Ciampa (Course Technology, Cengage Learning), *Security+ Guide to Network Security Fundamentals,* Third Edition, 2009.

RESEARCH AND BUILDING THE WEBSITE MONITORING SYSTEM FOR HUNG YEN UNIVERSITY OF TECHNOLOGY AND EDUCATION

Abstract:

In the report we present the process of designing, installing and testing a comprehensive security solution for the website of Hung Yen University of Technology and Education. We can test attack techniques and anti-hacking websites on the Internet. Research on the use of rabbin fingerprints for the Hung Yen University of Technology and Education, which can be extended to other websites on the Internet.

Keywords: *web security, Rabin fingerprint application, web protected.*