PHÁT HIỆN EMAIL URL LỪA ĐẢO SỬ DỤNG HỌC MÁY CÓ GIÁM SÁT

DETECT EMAIL URLS PHISHING USING SUPERVISED MACHINE LEARNING

Vũ Xuân Hạnh, Trần Tiến Dũng, Đỗ Thị Uyển, Hoàng Việt Trung, Ngô Minh Phương*

Ngày tòa soạn nhận được bài báo: 03/11/2021
Ngày nhận kết quả phản biện đánh giá: 03/05/2022
Ngày bài báo được duyệt đăng: 26/05/2022

Tóm tắt: *Cùng với tốc độ phát triển nhanh chóng của khoa học kỹ thuật và internet, các cuộc tấn công trên mạng ngày càng gia tăng với mức độ nguy hiểm cao và rất khó kiểm soát. Trong bài báo này, chúng tôi tập trung vào việc phát hiện email URL lừa đảo, là một dạng của các cuộc tấn công lừa đảo bằng cách đề xuất 51 đặc trưng URL để xác định. Chúng tôi sử dụng tập dữ liệu email URL Phishing có độ tin cậy cao và dựa trên các đặc trưng được trích chọn, nghiên cứu của chúng tôi đạt được độ chính xác tổng thể khoảng 94.53% khi sử dụng các kỹ thuật học máy có giám sát Random Forest.*

Từ khóa: *Tấn công URL Phishing, phát hiện Email URL Phishing, Học máy, Phát hiện tấn công lừa đảo qua thư, An ninh mạng, URL độc hại.*

Abstract: *Along with the rapid development of science and technology and the internet, cyber-attacks are increasing with high levels of danger and are difficult to control. In this paper, we focus on detecting email URL Phishing, which is a type of phishing attack by suggesting 51 URL features to identify. We use a highly reliable Phishing URL email dataset and based on the extracted features, our study achieves an overall accuracy of about 94.5% using supervisor machine learning Random Forest.*

Keywords: *Email URL Phishing, Detect Email URL Phishing, Machine Learning, Email URL Phishing attacks, URL Phishing, Cyber Security, Malicious URL.*

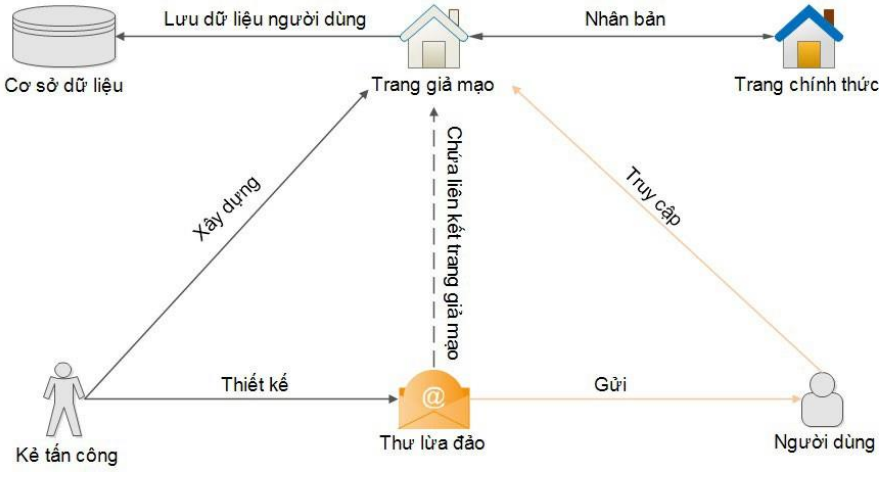
I. Đặt vấn đề

Thuật ngữ "lừa đảo" (*Phishing*), được dùng để chỉ các hành vi lừa đảo, đánh cắp tài khoản của người dùng Internet. Phishing là một kỹ thuật khiến người dùng hiểu lầm các URL mà họ truy cập là hợp pháp. Mục đích của hình thức lừa đảo này là thu thập các thông tin cá nhân như: thông tin đăng nhập, mật khẩu, thẻ tín dụng, thẻ ghi nợ hoặc tài khoản ngân hàng. Ngày nay, các cuộc tấn công lừa đảo ảnh hưởng rất nhiều đến các tổ chức tài

* Trường Đại học Mở Hà Nội

chính và cá nhân. Kẻ tấn công có thể ăn cắp thông tin qua thư điện tử, quảng cáo, trang web giả mạo,…

Đầu tiên, kẻ tấn công sẽ lựa chọn những trang chính thức có các giao dịch có liên quan đến thông tin cần đánh cắp. Sau đó, thực hiện hành vi nhân bản trang chính thức và xây dựng lại với ý đồ thu thập thông tin người dùng. Mặt khác, tạo email chứa liên kết tới trang giả mạo. Người dùng truy cập liên kết tới trang giả mạo, thực hiện giao dịch và từ đó thông tin bị đánh cắp lưu vào cơ sở dữ liệu của kẻ tấn công. Hình 1 mô tả quy trình tấn công email URL lừa đảo.



Hình 1: Tấn công Email URL lừa đảo

Có 316,747 cuộc tấn công trong tháng 10 năm 2021 được theo dõi bởi APWG [1], đây là số lượng cuộc tấn công lớn nhất trong lịch sử, cùng với đó, các cuộc tấn công cũng tăng gấp 3 so với đầu năm 2020. Trong số các email được báo cáo bởi người dùng doanh nghiệp, 51.8% là các cuộc tấn công lừa đảo đánh cắp thông tin xác thực. Sự gia tăng đáng kể này là một bằng chứng của sự tồn tại của các cuộc tấn công lừa đảo cùng với mức độ thiệt hại gia tăng mà chúng gây ra.

Trong bài báo này, chúng tôi sẽ đưa ra một giải pháp nhanh và hiệu quả để xác định email URL lừa đảo dựa trên các đặc trưng URL và tên miền trong URL. Trong phần còn lại của bài báo được cấu trúc như sau: mục II, chúng tôi thảo luận về một số nghiên cứu liên quan đến phát hiện URL lừa đảo, mục III trình bày về mô hình đề xuất, chi tiết về các đặc trưng trong URL và các chỉ số đánh giá. Kết quả thí nghiệm của chúng tôi được phân tích trong mục IV. Kết luận được trình bày trong mục V.

II. Cơ sở lý thuyết

Đã có nhiều công trình nghiên cứu đề xuất các kỹ thuật khác nhau để phát hiện các URL lừa đảo. Một trong số đó là việc duy trì một danh sách tên miền hoặc địa chỉ IP của các trang web lừa đảo đã được phát hiện trước đó. Một hệ thống có tên là Phishnet được đề xuất [2] là nơi duy trì một danh sách đen của các URL lừa đảo, hệ thống sẽ kiểm tra xem địa chỉ IP, tên máy chủ hoặc bản thân URL xem có thuộc danh sách đen đó hay không. Phương pháp duy trì danh sách trắng được đề xuất [3] có chứa tên miền và địa chỉ IP tương ứng của các trang web lành tính thay vì kỹ thuật

trên với danh sách đen. Phương pháp khai thác kết hợp quy tắc được đề xuất trong nghiên cứu của Jeeva và Rajsingh [4] để phát hiện các email URL lừa đảo và lành tính. Đối với phương pháp này, 14 đặc trưng khác nhau được trích chọn từ URL. Thuật toán TF-IDF được sử dụng để tìm các từ có tần suất cao trong các URL lừa đảo. Khoảng 93.00% URL lừa đảo được xác định chính xác bằng thuật toán Apriori trên tập dữ liệu gồm 1,400 URL.

Kenneth Fon Mbah trình bày trong luận văn thạc sỹ [5] đưa ra hệ thống cảnh báo lừa đảo (*PHAS*) có khả năng phát hiện và cảnh báo tất cả các loại email lừa đảo để giúp người dùng ra quyết định. Nghiên cứu này sử dụng tập dữ liệu email và dựa trên các đặc trưng được trích xuất, đề xuất đạt được độ chính xác khoảng 93.11% khi sử dụng các kỹ thuật máy học như: cây quyết định J48 và kNN. Shamal M. Firake[6] đề xuất một phương pháp để phát hiện và ngăn chặn các cuộc tấn công lừa đảo vào email.

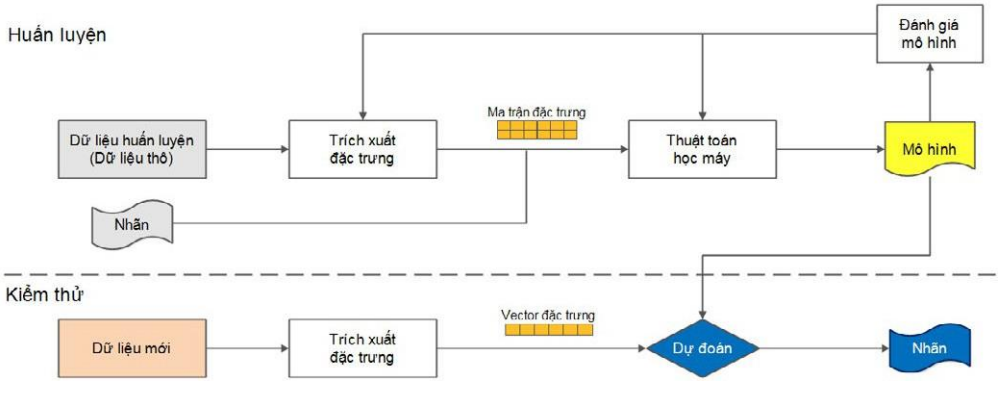
Các nghiên cứu trên hoạt động dựa trên danh sách tên miền, đặc trưng của URL, các đặc trưng khác được trích chọn từ trang web như WHOIS, công cụ tìm kiếm, v..v. Các nghiên cứu đã thu được những thành tựu như đã trình bày ở trên, tuy nhiên vẫn còn một số hạn chế: (i) Việc truy cập vào nội dung email để xác định URL lừa đảo dựa trên danh sách các URL lừa đảo hoặc URL hợp pháp không đáng tin cậy được duy trì tuy nhiên những kẻ tấn công có thể sử dụng các URL khác nhau cho mỗi lần tấn công; (ii) Trích chọn các đặc trưng cùng với sự trợ giúp của bên thứ 3 như WHOIS hoặc các công cụ tìm kiếm khác rất tốn thời gian; (iii) Chưa đề cập đến trích chọn các đặc trưng tên miền.

Nhằm tăng cao hiệu quả, chúng tôi đã xem xét các đặc trưng được trích chọn từ email URL lừa đảo và tên miền của URL để phát triển trong nghiên cứu.

III. Phương pháp nghiên cứu

3.1. Học máy có giám sát

Hình 2 mô tả kỹ thuật học máy có giám sát là nhóm các thuật toán dự đoán đầu ra (*outcome*) của một dữ liệu mới (*new input*) dựa trên các cặp (*input, outcome*) đã biết từ trước. Cặp dữ liệu này còn được gọi là (*dữ liệu, nhãn*). Đây là nhóm phổ biến nhất trong các thuật toán học máy. Thuật toán học máy có giám sát còn được tiếp tục chia nhỏ thành hai loại chính là: phân loại và hồi quy. Học máy có giám sát được sử dụng rộng rãi với bài toán phân loại nhị phân.



Hình 2: Mô hình học máy có giám sát

Trong kỹ thuật học máy có giám sát có một số thuật toán như: Naïve Bayes, kNN, cây quyết định J48, SVM, Random Forest…[7]

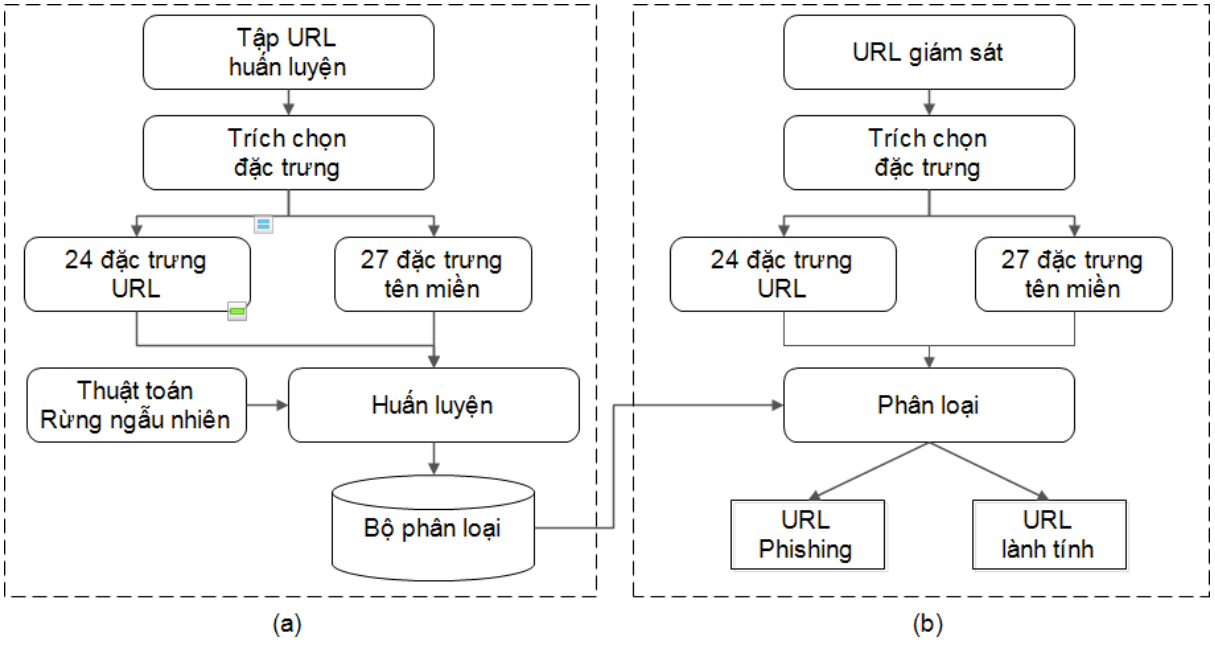
Thuật toán Random Forest xây dựng nhiều cây quyết định trên thuật toán cơ sở cây quyết định, tuy nhiên mỗi cây quyết định sẽ khác nhau (có yếu tố random). Sau đó kết quả dự đoán được tổng hợp từ các cây quyết định. Trong thuật toán cây quyết định, khi xây dựng cây quyết định nếu để độ sâu tùy ý thì cây sẽ phân loại đúng hết các dữ liệu trong tập huấn luyện dẫn đến mô hình có thể dự đoán tệ trên tập kiểm thử, khi đó mô hình sẽ có độ chính xác thấp. Tuy nhiên với thuật toán Random Forest mỗi cây lại có những yếu tố ngẫu nhiên: (i) Lấy ngẫu nhiên dữ liệu để xây dựng cây quyết định; (ii) Lấy ngẫu nhiên thuộc tính để xây dựng cây quyết định. Do mỗi cây quyết định trong thuật toán không dùng tất cả dữ liệu để huấn luyện, cũng như không dùng tất cả các thuộc tính của dữ liệu nên mỗi cây sẽ có dự đoán không tốt. Tuy nhiên, kết quả cuối cùng lại được tổng hợp từ nhiều cây quyết định nên thông tin từ các cây sẽ bổ sung cho nhau, dẫn đến mô hình sẽ có độ lệch và phương sai thấp, do đó mô hình sẽ có kết quả dự đoán tốt.

3.2. Mô hình phát hiện

Mô hình phát hiện email URL lừa đảo dựa trên máy học có giám sát đề xuất được chia thành 2 giai đoạn được minh họa tại hình 3 như sau:

(a) Giai đoạn huấn luyện: Tệp dữ liệu huấn luyện bao gồm email URL lừa đảo và lành tính. Các đặc trưng URL được trích chọn chia thành 2 loại: 24 đặc trưng URL và 27 đặc trưng tên miền. Sử dụng thuật toán Random Forest để huấn luyện, đưa ra bộ phân loại.

(b) Giai đoạn phát hiện: các URL được giám sát và trích chọn các đặc trưng, sử dụng bộ phân loại đã được huấn luyện để xác định email URL lừa đảo.



Hình 3: Mô hình phát hiện đề xuất

3.3. Trích chọn đặc trưng

3.3.1. Giới thiệu

Độ chính xác của hệ thống phát hiện email URL lừa đảo phụ thuộc vào các đặc trưng để phân biệt giữa các URL lừa đảo *và lành tính.* Trong các nghiên cứu gần đây, rất nhiều phân loại đặc trưng được lựa chọn như đặc trưng URL, đặc trưng mạng,...

Nghiên cứu *này tập trung vào các đặc trưng được trích chọn từ URL, chỉ cần xem xét URL mà không cần quan tâm đến các đặc trưng mạng,* các danh sách đã có trước... Các đặc trưng trích chọn từ nội dung web không được xem xét vì khi truy xuất nội dung trang web, những gói tin trong mạng có tải trọng lớn và tiêu tốn một lượng lớn tài nguyên để xử lý trong thời gian thực hoặc khi xử lý ngoại tuyến. Chúng tôi sử dụng 51 đặc trưng chia làm 2 nhóm để vector hoá các URL nhằm tăng hiệu quả của việc phát hiện, các đặc trưng được chia thành 2 nhóm như sau: (i) đặc trưng URL; (ii) đặc trưng tên miền.

3.3.2. Đặc trưng URL

Độ dài URL là một trong những đặc trưng đầu tiên [5], những kẻ tấn công sử dụng những URL có độ dài lớn để ẩn những phần đáng ngờ trong liên kết. Trong tệp dữ liệu huấn luyện của chúng tôi, độ dài trung bình của một email URL lừa đảo *là 63.13 ký tự*, trong khi đó với URL hợp pháp là 45.7 *ký tự*. Trong email URL lừa đảo thường có các ký tự các ký tự như ‘`’,’%’,’^’,’&’,’*’,’;’,… *là những ký tự đáng ngờ, và sự hiện diện của chúng xuất hiện nhiều hơn trong URL* lừa đảo. Một danh sách các từ đáng ngờ theo nghiên cứu [8] và với nhận định của chúng tôi, việc hiện diện của các từ này trong email URL lừa đảo nhiều hơn so với URL lành tính, bao gồm những từ như: ‘password’, ‘login’, ‘confirm’, ‘submit’, ‘payment’, ‘secure’, ‘account’, ‘index’, ‘token’, ‘signin’,… ngoài ra một số các từ đặc biệt mang tính chất nhạy cảm cũng xuất hiện trong các URL lừa đảo. Hiện nay có rất nhiều *công cụ hỗ trợ việc rút ngắn độ dài của URL. Và với các công cụ này, kẻ tấn công có thể che dấu được những đặc trưng dễ nhận biết trên URL đối với người dùng, và có thể đường dẫn đó là độc hại. Danh sách của các URL rút gọn này bao gồm: ‘bit\.ly’, ‘goo\.gl’, go2l*\.ink’, ‘x\.co’, ‘bitly\.com’, ’link\.zip\.net’. Đối với các URL lành tính việc xuất hiện của ký tự ‘.’ tương đối ít, thường là 1-2. Nhưng đối với các URL lừa đảo, số lượng này có thể là 4-5 hay thậm chí là 16. Điều này có liên quan đến các hostname chứa nhiều subdomain, do đó đường dẫn của URL lừa đảo *cũng sẽ dài hơn so với URL lành tính. Ngoài ra việc sử dụng các giao thức như: ‘HTTP’, ‘HTTPS’ và ‘FTP’ hoặc một vài giao thức khác. Theo như báo cáo của APWG* [1] việc sử dụng các giao thức như ‘HTTP’, ‘HTTPS’ đang có chiều hướng tăng lên ở những URL lừa đảo. Sự xuất hiện của địa chỉ IP, các dấu ‘\’, các cổng và chuyển hướng cũng được xem xét để trích chọn các đặc trưng URL [1][9][10]. Theo thống kê của chúng tôi, trong các URL lừa đảo thường chứa các chuỗi ký tự lớn hơn 30 ký tự (*chiếm 90% trong tổng số 155,996 URL*), đây được xem là một số khác biệt khá lớn đối với URL lành tính.

- f1: urlLength(u) – độ dài URL

- f2: tachar(u) - phân bố các ký tự đặc biệt trong URL

$$tachar(u) = \frac{countchar(u)}{len(u)} \quad (1)$$

trong đó, countchar(u) là số ký tự đặc biệt.

- f3: hasKeywords(u) - trả về giá trị là 1 nếu tồn tại các từ khóa, ngược lại trả về giá trị 0.

- f4: hasSpeChar(u) - trả về giá trị là 1 nếu tồn tại các từ khóa, ngược lại trả về giá trị 0.

- f5: hasSpeKW(u) - trả về giá trị là 1 nếu tồn tại từ nhạy cảm, ngược lại trả về giá trị 0.

- f6: tinyURL(u) - trả về giá trị là 1 nếu có URL rút gọn, ngược lại trả về giá trị 0.

- f7: tahex(u) - phân bố ký tự hexa trong URL

$$tahex(u) = \frac{counthe(u)}{len(u)} \quad (2)$$

trong đó, counthe(u) là số ký tự hexa.

-f8: tadigit(u) - phân bố chữ số trong URL

$$tadigit(u) = \frac{countdigit(u)}{len(u)} \quad (3)$$

trong đó, countdigit(u) là số chữ số.

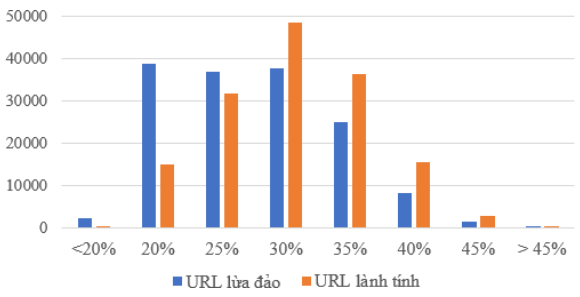
- f9: numdots(u) - số lượng các dấu '.' xuất hiện trong URL.

- f10: taslash(u) - phân bố dấu '/' trong URL

$$taslash(u) = \frac{countslash(u)}{len(u)} \quad (4)$$

trong đó, countslash(u) là số dấu '/'.

- f11: countcase(u) - số lượng các chữ in hoa



Hình 4: Tỷ lệ phân bố nguyên âm trong URL

Tỷ lệ phân bố nguyên âm trong URL lừa đảo và lành tính được thể hiện trong hình 4 cho thấy có sự khác biệt, do đó các đặc trưng f12, f13 được bổ sung trong nghiên cứu này.

- f12: numvo(u)* - phân bố nguyên âm trong URL.

$$numvo(u) = \frac{countvo(u)}{len(u)} \quad (5)$$

trong đó, countvo(u) là số nguyên âm.

- f13: numco(u)* - phân bố phụ âm trong URL.

$$numco(u) = \frac{countco(u)}{len(u)} \quad (6)$$

trong đó, countvo(u) là số phụ âm.

- f14: numsdm(u) - số lượng các subdomain.

- f15: radomain(u) - tỉ lệ độ dài của domain so với URL.

$$radomain(u) = \frac{lend(u)}{len(u)} \quad (7)$$

trong đó, lend(u) là độ dài domain.

- f16: rapath(u) - Tỉ lệ độ dài của đường dẫn so với URL.

$$rapath(u) = \frac{lenpath(u)}{len(u)} \quad (8)$$

trong đó, lenpath(u) là độ dài domain

- f17: haspro(u) - trả về giá trị 1 nếu tồn tại 'http', 'https', 'www'trong URL, ngược lại trả về giá trị 0.

- f18: hasIP(u) - trả về giá trị 1 nếu tồn tại địa chỉ IP trong URL, ngược lại trả về giá trị 0.

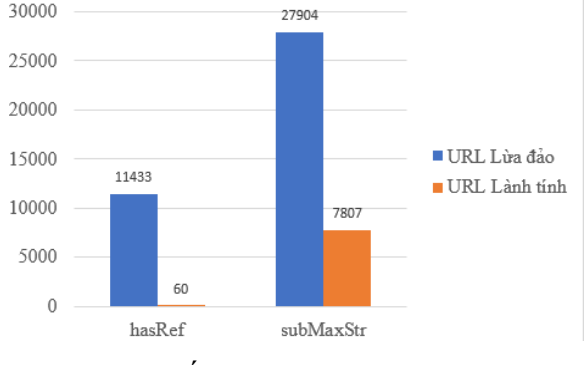
- f19: hasExe(u) - trả về giá trị 1 nếu tồn tại file có phần mở rộng '.exe', ngược lại trả về giá trị 0.

- f20: hasport(u) - trả về giá trị 1 nếu tồn tại cổng trong URL, ngược lại trả về giá trị 0.

- f21: backslash(u) - trả về giá trị 1 nếu tồn tại dấu '\' trong URL, ngược lại trả về giá trị 0.

- f22: redirect(u) - trả về giá trị là 1 nếu tồn tại chuyển hướng trong URL, ngược lại trả về giá trị 0.

Thống kê trong 150,000 URL lừa đảo và 150,000 URL lành tính. Kết quả được thể hiện trong hình 5 cho thấy: các cụm 'ref=', 'cdm=' trong URL lừa đảo xuất hiện nhiều hơn (11433 lần) so với URL lành tính (60 lần). Tương tự, các chuỗi ký tự dài (>25 ký tự) trong các URL lừa đảo cũng xuất hiện nhiều hơn gấp 4 lần so với URL lành tính. Đây là lý do chúng tôi bổ sung 2 đặc trưng mới f23 và f24.



Hình 5: Thống kê hasRef và subMaxStr

- f23: hasref(u)* - trả về giá trị là 1 nếu tồn tại các cụm 'ref=', 'cdm=' …trong URL, ngược lại trả về giá trị 0.

- f24: maxsub30(u)* - trả về giá trị là 1 nếu chuỗi con lớn nhất có độ dài lớn hơn 30 ký tự, ngược lại trả về giá trị 0.

3.3.3. Đặc trưng tên miền

Kế thừa nghiên cứu trước đây của chúng tôi [9] [10], bi-gram là một cụm gồm 2 ký tự kề nhau được trích ra từ một chuỗi ký tự. Ví dụ, với chuỗi "domain" gồm các bi-gram: do, om, ma, ai, in. Một tên miền có thể chứa các ký tự trong tập 26 ký tự chữ cái (a-z), các ký tự số (0-9), ký tự "." và "-", do đó tổng số bi-gram là S(bi-gram) = 38^2=1,444. Tương tự, tri-gram là một cụm gồm 3 ký tự kề nhau được trích ra từ một chuỗi ký tự. Với ví dụ trên ta có các tri-gram: dom, oma, mai, ain và tổng số tri-gram là S(tri-gram) = 38^3=54,872. Từ tập hợp các tên miền lành tính được trích từ top 100,000 tên miền trên Alexa [11] rút ra danh sách gồm K=1,000 cụm n-gram có tần suất xuất hiện cao nhất, ký hiệu DS(n-gram). DS(n-gram) được sử dụng cho việc tính toán 8 đặc trưng bi-gram (f25 - f32) và 8 đặc trưng (f33 - f40) tri-gram. Ngoài ra, chúng tôi cũng sử dụng các đặc trưng thống kê như: tỷ lệ nguyên âm, tỷ lệ phụ âm, tỷ lệ ký tự '-','.' và chữ số trong tên miền. Hơn nữa, đối với các tên miền lành tính thường được sinh ra dựa trên các nguyên tắc sử dụng từ trong ngôn ngữ tự nhiên. Bảng 1 liệt kê xác suất xuất hiện của các chữ cái trong 100,000 tên miền lành tính để tính EOD cho từng tên miền. 27 đặc trưng n-gram và thống kê của tên miền trong URL được liệt kê dưới đây.

Bảng 1: Xác suất của 38 ký tự trong 100.000 tên miền

C	P(C)	C	P(C)	C	P(C)	C	P(C)	C	P(C)	C	P(C)
a	9.35	g	2.40	m	3.37	s	6.48	y	1.67	5	0.10
b	2.27	h	2.56	n	6.12	t	6.13	x	0.68	6	0.09
c	3.87	i	7.40	o	7.28	u	3.23	0	0.18	7	0.09
d	3.26	j	0.55	p	2.91	v	1.37	1	0.24	8	0.10
e	9.69	k	1.90	q	0.21	w	1.20	2	0.23	9	0.08
f	1.67	l	4.56	r	6.44	x	0.67	3	0.15	.	0.00
								4	0.16	-	1.26

- f25-f33: count(d) - số lượng n-gram của tên miền d.- f26-f34: m(d) -là phân bố tần suất chung của các n-gram trong tên miền d.

$$m(d) = \sum_{i=1}^{count(d)} f(i) * index(i) \qquad (9)$$

trong đó f(i) là tổng số lần xuất hiện của n-gram i trong DS(n-gram) và index(i) là thứ hạng của n-gram i trong TS(n-gram)

- f27-f35: s(d) - là trọng số n-gram.

$$s(d) = \frac{\sum_{i=1}^{count(d)} f(i) * vt(i)}{count(d)} \qquad (10)$$

trong đó, vt(i) là thứ hạng củan-gram i trong DS(n-gram).

- f28-f36: ma(d) - là trung bình phân bố tần suất chung của các n-gram của tên miền d.

$$ma(d) = \frac{m(d)}{sum\_ng(d)} \qquad (11)$$

len(d) là tổng số các n-gram có trong tên miền d.

- f29-f37: sa(d) - là trung bình trọng số n-gram của tên miền d.

$$sa(d) = \frac{s(d)}{sum\_ng(d)} \qquad (12)$$

- f30-f38: tan(d) - là trung bình số lượng n-gram phổ biến của tên miền d.

$$tan(d) = \frac{count(d)}{sum\_ng(d)} \qquad (13)$$

- f31-f39: taf(d) - là trung bình tần suất n-gram phổ biến của tên miền d.

$$taf(d) = \frac{\sum_{i=1}^{count(d)} f(i)}{sum\_ng(d)} \qquad (14)$$

- f32-f40: là entropy của tên miền d.

$$ent(d) = -\sum_{i=1}^{count(d)} \frac{vt(i)}{K} * log(\frac{vt(i)}{K}) \qquad (15)$$

K là số cụm n-gram phổ biến

- f41: tanv(d) - là phân bố nguyên âm của tên miền d.

$$tanv(d) = \frac{countnv(d)}{len(d)} \qquad (16)$$

countnv(d) là số nguyên âm, len(d) là số ký tự của tên miền d.

- f42: tanco(d) - là phân bố phụ âm của tên miền d.

$$tanco(d) = \frac{countco(d)}{len(d)} \qquad (17)$$

countco(d) là số phụ âm của tên miền d.

- f43: tandi(d) - là phân bố chữ số của tên miền d.

$$tanco(d) = \frac{countdi(d)}{len(d)} \quad (18)$$

countdi(d) là số chữ số của tên miền d.

- f44: tansc(d) - là phân bố ký tự đặc biệt của tên miền d.

$$tansc(d) = \frac{countsc(d)}{len(d)} \quad (19)$$

countsc(d) là số ký tự đặc biệt

- f45: tanhe(d) - là phân bố ký tự hexa của tên miền d.

$$tanhe(d) = \frac{counthe(d)}{len(d)} \quad (20)$$

counthe(d) là số ký tự hexa của tên miền d

- f46: is_digit(d) - trả về giá trị 1 nếu ký tự đầu tiên của tên miền d là số, ngược lại trả về giá trị 0.

- f47: len(d) - độ dài tên miền d.

- f48: ent_char(d) - là entropy của miền d. D(x) là phân phối xác suất của ký tự x trong miền d.

$$ent\_char(d) = -\frac{\sum_x D(x)\log(D(x))}{\log(len(d))} \quad (21)$$

- f49: EOD(d) - là giá trị kỳ vọng của tên miền d. Tên miền bao gồm k ký tự $\{x_1, x_2,...,x_k\}$. $n(x_i)$ là tần suất xuất hiện của ký tự x_i và $p(x_i)$ là phân phối xác suất của ký tự x_i. được tính bằng cách sử dụng top 100,000 tên miền được liệt kê bởi Alexa, EOD(d).

$$EOD(d) = \frac{\sum_{i=1}^{k} n(x_i)\,p(x_i)}{\sum_{i=1}^{k} n(x_i)} \quad (22)$$

Đối với các tên miền lừa đảo, kẻ tấn công thường sử dụng kỹ thuật sinh tự động nên thông thường các tên miền này không xuất hiện trong rank Alexa.

- f50: rank(d)* - xếp hạng domain trong danh sách Alexa.

Thống kê ra top5 các TLD được sử dụng trong 156,000 URL lành tính (*chiếm xấp xỉ 92%*) trong khi đó các URL lừa đảo sử dụng TLD rất đa dạng. Do đó đặc trưng TLD của các email URL được xem xét để sử dụng trong nghiên cứu này.

- f51: tld(d)* - trả về giá trị 1 nếu TLD trong top5 LTD lành tính, ngược lại trả về giá trị 0.

3.3.4. Phương pháp đánh giá

- Để đánh giá mô hình đề xuất, sử dụng sáu độ đo bao gồm: PPV, TPR, FPR, FNR, F1 và ACC. Các độ đo được tính toán như sau:

Độ chính xác (PPV-Positive Predictive Value) được tính theo công thức:

$$PPV = \frac{TP}{TP + FP} \quad (23)$$

Tỷ lệ dương tính đúng (TPR), hay độ nhạy, được tính theo công thức:

$$TPR = \frac{TP}{TP + FN} \quad (24)$$

Tỷ lệ dương tính giả (FPR) hay còn gọi "nhầm lẫn", được tính theo công thức:

$$FPR = \frac{FP}{FP + TN} \quad (25)$$

Tỷ lệ âm tính giả (FPR) hay còn gọi "bỏ sót", được tính theo công thức:

$$FNR = \frac{FN}{FN + TP} \quad (26)$$

Độ đo F1 được tính theo công thức:

$$F1=\frac{2TP}{2TP+FP+FN} \quad (27)$$

Độ chính xác toàn cục, hay độ chính xác chung ACC, được tính theo công thức:

$$ACC=\frac{TP+TN}{TP+TN+FP+FN} \quad (28)$$

trong đó, TP là số lượng các URL lừa đảo được phân loại đúng, TN là số lượng các URL lành tính được phân loại đúng, FP là số lượng URL lành tính bị phân loại sai thành URL lừa đảo và FN là số lượng các URL lừa đảo bị phân loại sai URL lành tính.

IV. Kết quả và thảo luận

4.1. Tập dữ liệu huấn luyện và kiểm thử

Để đánh giá độ chính xác phân loại email URL lừa đảo và lành tính sử dụng học máy, sử dụng các tập dữ liệu tên miền đã được bóc tách và gán nhãn [12], bao gồm tập các email URL lừa đảo và lành tính. Các email URL lành tính được gán nhãn 0 và email URL lừa đảo được gán nhãn 1.

Bảng 2: Dữ liệu huấn luyện và kiểm thử

Tập dữ liệu huấn luyện và kiểm thử	Email URL	
	Lành tính	*Phising*
Tập dữ liệu huấn luyện	100,000	100,000
Tập kiểm thử Dataset 1		20,000
Tập kiểm thử Dataset 2		35,996

4.2. Lựa chọn thuật toán

Với tập dữ liệu huấn luyện, sử dụng một số thuật toán học máy kiểm tra chéo 10 lần để xác định hiệu suất của mô hình. Dựa vào kết quả tại Bảng 3, với ACC và F1 lần lượt bằng 94.50% và 94.54% kèm theo tỷ lệ âm tính giả và dương tính giả là 4.73% và 6.27% thuật toán RF cho hiệu quả tốt nhất. Mặt khác, thử nghiệm RF với lần lượt 40, 45, 50, 55 cây được ACC lần lượt là: 94.44%, 94.41%, 94.50%, 94.48% . Do đó, chúng tôi lựa chọn thuật toán Random Forest với số cây là 50 ***để*** huấn luyện ***mô hình và kiểm thử.***

Bảng 3: Hiệu suất của một số kỹ thuật học máy

Thuật toán	ACC	F1
Random Forest	94.50%	94.54%
Logistic	84.47%	84.61%
J48	91.81%	91.80%
Naïve Bayes	81.63%	81.63%
kNN	91.86%	91.80%

Mặt khác, để so sánh và làm rõ hiệu quả của mô hình khi thêm 6 đặc trưng mới bổ sung vào 45 đặc trưng đã kế thừa cho kết quả như bảng 4. Khi thêm 6 đặc trưng mới, độ chính xác toàn cục tăng 0.98%, tỷ lệ tăng không cao do tỷ lệ ACC tới ngưỡng khả năng cải thiện hiệu suất của mô hình là rất thấp. Tuy nhiên, tỷ lệ âm tính giả giảm đi đáng kể từ 6.19% tới 4.73%, tỷ lệ bỏ sót giảm tức là hiệu suất của mô hình tốt hơn.

Bảng 4: So sánh mô hình 45 và 51 đặc trưng

Đặc trưng	FNR	FPR	ACC
45	6.19%	6.69%	93.56%
51	4.73%	6.27%	94.50%

4.3 Kết quả và đánh giá

Sử dụng mô hình đề xuất với thuật toán RF sử dụng 50 cây kiểm thử 02 tệp dữ liệu dataset1 và dataset2 cho kết quả lần lượt là 95.63% và 95.51% được thể hiện tại Bảng 5.

Bảng 5: Hiệu suất kiểm thử

Tệp	Số lượng	Phát hiện	Tỷ lệ
Dataset1	20,000	19,127	95.63%
Dataset2	35,996	34,383	95.51%

Bảng 6: So sánh các đề xuất

Đề xuất	**Sử dụng**	**Tỷ lệ**
Jeeva và cộng sự [5]	Apriori	93.00%
Kenneth [4]	J48	93.11%
Của chúng tôi	RF (50)	94.50%

Từ kết quả huấn luyện mô hình, so sánh với một số nghiên cứu trước được thể hiện tại Bảng 6 cho thấy mô hình của chúng tôi có hiệu suất cao hơn. Tuy nhiên, Jeeva và cộng sự sử dụng khai phá luật kết hợp Apriori, Kenneth sử dụng J48 với các bộ dữ liệu khác nhau. Do đó, việc so sánh chưa được tuyệt đối chính xác.

V. Kết luận

Với mục đích hạn chế các cuộc tấn công trên mạng nói chung và các cuộc tấn công URL lừa đảo nói riêng. Chúng tôi đã nghiên cứu chi tiết các đặc trưng của URL và tên miền trong URL. Ngoài các đặc trưng kế thừa từ các nghiên cứu trước đây của các tác giả khác và của chúng tôi, trong bài báo này chúng tôi đề xuất thêm một số đặc trưng mới, cụ thể là các đặc trưng: f12, f13, f22, f30, f50 và f51. Từ kết quả nghiên cứu trên, chúng tôi đề xuất mô hình phát hiện email URL lừa đảo dựa trên đặc trưng URL và tên miền chứa trong URL. Trong nghiên cứu này, chúng tôi là xây dựng một phương pháp phát hiện email URL lừa đảo nhanh chóng, hiệu quả và không phụ thuộc vào các đặc trưng mạng cũng như hiệu suất của thiết bị cụ thể kết quả được trình bày tại mục 4.3.

Trong tương lai, chúng tôi tiếp tục nghiên cứu các bộ đặc trưng khác nhau và sử dụng các tập dữ liệu lớn hơn để giúp phát hiện email URL lừa đảo chính xác và hiệu quả hơn.

Tài liệu tham khảo:

[1]. "*Phishing Activity Trends Reports*", https://apwg.org/trendsreports/. Truy cập 1-2022

[2]. Pawan P và cộng sự, "*Predictive Blacklisting to Detect Phishing Attacks*", p:1-5, Proceedings IEEE INFOCOM, 2010.

[3]. Jain, A. K., & Gupta, B. B. "*A novel approach to protect against phishing attacks at client side using autoupdated white-list*". EURASIP Journal on Information Security,2016(1). doi:10.1186/s13635-016-0034-3, 2016

[4]. Jeeva, S. C., & Rajsingh, E. B. "*Intelligent phishing url detection using association rule mining*". Humancentric Computing and Information Sciences ,6(1).

[5]. doi:10.1186/s13673-016-0064-3, 2016.

[6]. Kenneth Fon, Arash Habibi Lashkari Ali A. Ghorbani. "*A phishing Email Detection Approach Using Machine Learing Techniques*", Innsbruck, Austria, January 26-27, 2017

[7]. Shamal M. Firake, Pravin Soni and B.B. Meshram, "*Tool For Prevention and Detection of Phishing E-mail Attacks*", Computer technology Department, V.J.T.I. , Matunga, Mumbai. 2011.

[8]. Tiep, V.H., "*Machine Learning cơ bản*". 2016-2020.

[9]. Bahnsen, A. C., Bohorquez, E. C., Villegas, S., Vargas, J., & Gonzalez, F. A. "*Classifying phishing URLs using recurrent neural networks*". 2017 APWG Symposium on Electronic Crime Research (eCrime). doi:10.1109/ecrime.2017.7945048, 2017

[10]. Xuan Dau Hoang and Xuan Hanh Vu, "*An Improved Model For Detecting DGA Botnets Using Random Forest Algorithmm*", 2021; DOI: 10.1080/19393555.2021.1934198

[11]. Hoang X.D. and Nguyen Q.C, "*Botnet Detection Based On Machine Learning Techniques Using DNS Query Data*", Future Internet, 2018, 10, 43; doi:10.3390/fi10050043.

[12]. Alexa. *Alexa Top 1M.* [cited 2019;Available from: http://s3.amazonaws.com/alexa-static/

[13]. Tarun Tiwari, *Phishing Site URLs Dataset*, https://www.kaggle.com/taruntiwarihp/phishing-site-urls

Địa chỉ tác giả: Trường Đại học Mở Hà Nội

Email: hanhvx@hou.edu.vn