TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &  
TRUYỀN THÔNG VIỆT - HÀN

**Khoa Khoa Học Máy Tính**



BÀI TẬP LỚN

**Tìm hiểu về trình đọc file PDF và khai thác các lỗ hổng của trình đọc file PDF (Client-side)**

Sinh viên: **NGUYỄN AN PHÚC** Mã: **21IT639**

**TRẦN VŨ LẬP THI** Mã: **21IT448**

Giảng viên hướng dẫn: **TS. Trần Thế Sơn**

Đà Nẵng, tháng 11 năm 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &  
TRUYỀN THÔNG VIỆT - HÀN

**Khoa Khoa Học Máy Tính**



BÀI TẬP LỚN

**Tìm hiểu về trình đọc file PDF và khai thác các lỗ hổng của trình đọc file PDF (Client-side)**

Sinh viên: **NGUYỄN AN PHÚC** Mã: **21IT639**

**TRẦN VŨ LẬP THI** Mã: **21IT448**

Giảng viên hướng dẫn: **TS. Trần Thế Sơn**

Đà Nẵng, tháng 11 năm 2024

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

**…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

LỜI CẢM ƠN

Để đề tài đạt được kết quả như hiện tại, chúng em đã nhận được sự hỗ trợ,   
giúp đỡ, chỉ bảo tận tâm của thầy. Với tình cảm sâu sắc, chân thành, cho phép   
chúng em được bày tỏ lòng biết ơn sâu sắc đến tất cả quý thầy cô đã tạo điều kiện   
giúp đỡ chúng em trong quá trình học tập và nghiên cứu.

Trước hết, chúng em xin gửi tới đội ngũ giảng viên trường Đại học Công nghệ thông tin và truyền thông Việt – Hàn lời chào trân trọng, lời chúc sức khỏe và   
lời cảm ơn sâu sắc nhất. Với sự quan tâm, chỉ bảo tận tình chu đáo, cũng như vốn  
kiến thức về nhiều lĩnh vực của mình, đội ngũ giảng viên của trường đã truyền tải cho chúng em được nhiều kiến thức, cũng như kỹ năng cần thiết, để đến ngày hôm nay,  
chúng em đã có thể hoàn thành được Đề tài của mình.

Đặc biệt, chúng em xin gửi lời cảm ơn chân thành tới TS.Trần Thế Sơn,   
giảng viên đã quan tâm, giúp đỡ, chỉ bảo tận tình để chúng em có thể hoàn thành tốt   
đề tài này trong suốt thời gian vừa qua.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế, đề tài này không thể tránh được những thiếu sót. Chúng em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy/cô, để chúng em có điều kiện bổ sung, nâng cao kiến thức, kỹ năng   
của mình, phục vụ tốt hơn cho công tác thực tế sau này.

Chúng em xin trân trọng cảm ơn!

*Sinh viên,*

Nguyễn An Phúc,

Trần Vũ Lập Thi.

**MỤC LỤC**

[LỜI CẢM ƠN ii](#_Toc3911)

[DANH MỤC HÌNH ẢNH vi](#_Toc21334)

[DANH MỤC CÁC TỪ VIẾT TẮT viii](#_Toc4695)

[MỞ ĐẦU x](#_Toc6844)

[1. Giới thiệu x](#_Toc15489)

[2. Mục tiêu của đề tài x](#_Toc17602)

[3. Đối tượng, phạm vi và phương pháp tiếp cận xi](#_Toc18427)

[4. Đóng góp của đề tài xi](#_Toc31821)

[5. Bố cục của đề tài xii](#_Toc12177)

[CHƯƠNG 1: TỔNG QUAN VỀ CÔNG CỤ VÀ MÔI TRƯỜNG PHÁT TRIỂN 1](#_Toc3669)

[1.1. Khái niệm về PDF và trình đọc file PDF 1](#_Toc2907)

[1.1.1. Giới thiệu về PDF 1](#_Toc21449)

[1.1.2. Khái niệm trình đọc PDF 1](#_Toc16044)

[1.2. Lịch sử phát triển của PDF và các trình đọc PDF 2](#_Toc26989)

[1.2.1. Sự ra đời và phát triển của định dạng PDF 2](#_Toc16160)

[1.2.2. Sự phát triển của các trình đọc PDF 3](#_Toc13202)

[1.3. Các trình đọc PDF phổ biến 3](#_Toc20298)

[1.3.1. Adobe Acrobat Reader 3](#_Toc17379)

[1.3.2. Foxit Reader 4](#_Toc13834)

[1.3.3. Trình đọc PDF tích hợp trên trình duyệt 5](#_Toc14900)

[1.4. Cấu trúc của file PDF 5](#_Toc32389)

[1.4.1. Header 5](#_Toc23831)

[1.4.2. Body 5](#_Toc9477)

[1.4.3. Cross-reference Table 6](#_Toc12220)

[1.4.4. Trailer 6](#_Toc32705)

[1.5. Các công cụ và môi trường phát triển, kiểm tra trình đọc PDF 6](#_Toc14372)

[1.5.1. Công cụ phát triển 6](#_Toc27345)

[1.5.2. Các công cụ kiểm tra bảo mật PDF 7](#_Toc32699)

[1.6. Metasploit 8](#_Toc14558)

[1.7. Tổng kết 8](#_Toc27075)

[CHƯƠNG 2: Các lỗ hổng tồn tại và phương thức khai thác trên trình đọc PDF 9](#_Toc13909)

[2.1. Tổng quan về lỗ hổng trên trình đọc PDF 9](#_Toc5705)

[2.1.1. Định nghĩa và tầm quan trọng của bảo mật trong trình đọc PDF 9](#_Toc15138)

[2.1.2. Các loại lỗ hổng phổ biến trên trình đọc PDF 9](#_Toc32109)

[2.2. Lỗ hổng thực thi mã từ xa (Remote Code Execution - RCE) 11](#_Toc26423)

[2.2.1. Cách thức hoạt động của lỗ hổng RCE 11](#_Toc3657)

[2.2.2. Ví dụ về khai thác lỗ hổng RCE 12](#_Toc18547)

[2.3. Lỗ hổng XSS (Cross-Site Scripting) trên trình đọc PDF 14](#_Toc13760)

[2.3.1. Cách thức hoạt động của lỗ hổng XSS trong PDF 14](#_Toc23721)

[2.3.2. Ví dụ về khai thác XSS trên PDF 15](#_Toc5359)

[2.4. Lỗ hổng lừa đảo (Phishing) qua PDF 16](#_Toc14538)

[2.4.1. Phương thức hoạt động của lỗ hổng lừa đảo qua PDF 16](#_Toc11300)

[2.4.2. Ví dụ về lừa đảo qua PDF 17](#_Toc9862)

[2.5. Lỗ hổng tiết lộ thông tin (Information Disclosure) 19](#_Toc9856)

[2.5.1. Cách thức hoạt động của lỗ hổng tiết lộ thông tin 19](#_Toc21476)

[2.5.2. Ví dụ về khai thác lỗ hổng tiết lộ thông tin 20](#_Toc5038)

[2.6. Các công cụ và phương pháp kiểm tra lỗ hổng trên trình đọc PDF 22](#_Toc19613)

[2.6.1. Công cụ phát hiện và kiểm tra lỗ hổng 22](#_Toc2137)

[2.6.2. Phương pháp kiểm tra lỗ hổng bảo mật trên trình đọc PDF 23](#_Toc29499)

[2.7. Tổng kết 24](#_Toc9029)

[Kết luận 25](#_Toc8269)

[CHƯƠNG 3: MÔ PHỎNG 26](#_Toc9267)

[3.1. Mô hình hệ thống 26](#_Toc52)

[3.2. Chuẩn bị môi trường 26](#_Toc27395)

[3.3. Mô phỏng lỗ hỏng Remote Code Execution 27](#_Toc19538)

[3.4. Mô phỏng lỗ hỏng XSS 30](#_Toc10405)

[KẾT LUẬN 33](#_Toc8858)

[1. Kết quả đạt được 33](#_Toc6365)

[2. Những mặt đạt được và chưa đạt được 33](#_Toc5854)

[TÀI LIỆU THAM KHẢO 35](#_Toc4842)

DANH MỤC HÌNH ẢNH

[Hình 1.1.1 : Biểu tượng của định dạng file PDF 1](#_Toc28807)

[Hình 1.2.1 : Quá trình phát triển của PDF qua các mốc thời gian 2](#_Toc1509)

[Hình 1.2.2 : Giao diện của các trình đọc PDF phổ biến 3](#_Toc31378)

[Hình 1.3.1 : Adobe Acrobat Reader 4](#_Toc4706)

[Hình 1.3.2 : Foxit Reader 4](#_Toc11992)

[Hình 1.4.1 : Cấu trúc của file PDF 5](#_Toc21663)

[Hình 1.5.1 : Công cụ PDF JS 6](#_Toc7705)

[Hình 1.5.2 : Công cụ MuPDF. 7](#_Toc2827)

[Hình 1.5.3 : Burp Suite phân tích bảo mật PDF 7](#_Toc24294)

[Hình 1.6.1 : Metasploit 8](#_Toc2637)

[Hình 2.1.1 : Tấn công phishing qua PDF 10](#_Toc16830)

[Hình 2.2.1 : Cơ chế hoạt động của lỗ hổng thực thi mã từ xa trên file PDF 11](#_Toc12936)

[Hình 2.2.2 : Lỗ hổng CVE-2018-4990 12](#_Toc4216)

[Hình 2.2.3 : Ví dụ về khai thác lỗ hổng thực thi mã từ xa trên file PDF 13](#_Toc18910)

[Hình 2.3.1 : Cơ chế của lỗ hổng XSS trên trình đọc PDF 14](#_Toc23776)

[Hình 2.3.2 : Ví dụ khai thác lỗ hổng XSS trên PDF 15](#_Toc24627)

[Hình 2.4.1 : Phương thức lừa đảo qua file PDF 17](#_Toc8869)

[Hình 2.5.1 : Lỗ hổng tiết lộ thông tin trong metadata của file PDF 20](#_Toc8243)

[Hình 2.6.1 : Các công cụ phân tích và kiểm tra lỗ hổng trên PDF 22](#_Toc16499)

[Hình 2.6.2 : Quy trình kiểm tra bảo mật cho file PDF 24](#_Toc12694)

[Hình 2.7.1 : Tổng kết các lỗ hổng bảo mật trên trình đọc PDF 25](#_Toc3164)

[Hình 3.1.1 : Sơ đồ hệ thống 26](#_Toc4258)

[Hình 3.2.1 : Kali Linux 26](#_Toc1916)

[Hình 3.2.2 : Windows 7 26](#_Toc32498)

[Hình 3.2.3 : Adobe reader 9 27](#_Toc29917)

[Hình 3.3.1 : cấu hình tạo file PDF độc hại 27](#_Toc29047)

[Hình 3.3.2 : Tạo thành công file PDF độc hại 27](#_Toc22424)

[Hình 3.3.3 : Thiết lập listener. 28](#_Toc8124)

[Hình 3.3.4 : Cấu hình các thông số cho listener 28](#_Toc7229)

[Hình 3.3.5 : Chạy listener 28](#_Toc6254)

[Hình 3.3.6 : phát tán file qua web 28](#_Toc29045)

[Hình 3.3.7 : Máy nạn nhân tải file độc hại về 29](#_Toc32539)

[Hình 3.3.8 : Máy tấn công đã kết nối được với máy nạn nhân 29](#_Toc5094)

[Hình 3.3.9 : Chụp màn hình máy nạn nhân 29](#_Toc22330)

[Hình 3.3.10 Theo dõi hành động trực tiếp trên máy nạn nhân 30](#_Toc4439)

[Hình 3.4.1 : Tạo file PDF 30](#_Toc108)

[Hình 3.4.2 : Tạo file JavaScipt 31](#_Toc18197)

[Hình 3.4.3 : Nội dung file JS 31](#_Toc18433)

[Hình 3.4.4 : File PDF đã nhúng JavaScript thành công 31](#_Toc8085)

[Hình 3.4.5 : Phát tán file độc hại qua Web 31](#_Toc13246)

[Hình 3.4.6 : File chứa JS đã được thực thi 32](#_Toc21338)

DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **TỪ VIẾT TẮT** | **ĐẦY ĐỦ** | **NỘI DUNG** |
| RCE | Remote Code Execution | Thực thi mã từ xa, một lỗ hổng cho phép kẻ tấn công chạy mã độc trên hệ thống của nạn nhân. |
| XSS | Cross-Site Scripting | Lỗ hổng cho phép chèn mã JavaScript độc hại vào tài liệu PDF và thực thi trên trình duyệt của người dùng. |
| PDF | Portable Document Format | Định dạng tài liệu di động, một định dạng tệp phổ biến được sử dụng để trình bày tài liệu. |
| CVE | Common Vulnerabilities and Exposures | Danh sách các lỗ hổng bảo mật đã biết công khai. |
| SAST | Static Application Security Testing | Kiểm thử bảo mật ứng dụng tĩnh, một phương pháp kiểm thử bảo mật trong quá trình phát triển mã nguồn. |
| 2FA | Two-Factor Authentication | Xác thực hai yếu tố, một phương pháp bảo mật yêu cầu hai yếu tố xác thực để xác minh danh tính người dùng. |
| SQLi | SQL Injection | Tiêm mã SQL, một lỗ hổng bảo mật cho phép kẻ tấn công chèn mã SQL vào truy vấn để truy cập dữ liệu trong cơ sở dữ liệu. |
| CSRF | Cross-Site Request Forgery | Giả mạo yêu cầu từ trang khác, một lỗ hổng bảo mật cho phép kẻ tấn công thực hiện các hành động trái phép thay cho người dùng. |
| UAC | User Account Control | Kiểm soát tài khoản người dùng, một tính năng bảo mật trên Windows để ngăn chặn các thay đổi trái phép đối với hệ thống. |
| CVSS | Common Vulnerability Scoring System | Hệ thống chấm điểm lỗ hổng chung, một tiêu chuẩn để đánh giá mức độ nghiêm trọng của các lỗ hổng bảo mật. |
| IDS | Intrusion Detection System | Hệ thống phát hiện xâm nhập, một hệ thống giám sát mạng hoặc hệ thống để phát hiện các hành vi đáng ngờ hoặc các vi phạm chính sách. |

MỞ ĐẦU

* 1. Giới thiệu

Trong thời đại số hóa và liên kết toàn cầu ngày nay, tài liệu kỹ thuật số đóng vai trò quan trọng trong việc truyền tải và lưu trữ thông tin. Định dạng PDF (Portable Document Format) là một trong những định dạng tài liệu phổ biến nhất, được sử dụng rộng rãi trong nhiều lĩnh vực từ giáo dục, y tế, kinh doanh đến quản lý nhà nước. Để xem và tương tác với các file PDF, người dùng cần sử dụng các trình đọc PDF như Adobe Reader, Foxit Reader, và nhiều phần mềm khác. Tuy nhiên, các trình đọc PDF không chỉ là công cụ hữu ích mà còn là mục tiêu tiềm năng của các cuộc tấn công từ phía các hacker.

Việc khai thác các lỗ hổng bảo mật trong trình đọc PDF là một mối đe dọa nghiêm trọng, có thể gây ra rò rỉ thông tin, mất mát dữ liệu và thậm chí là chiếm quyền điều khiển hệ thống của người dùng. Do đó, việc tìm hiểu và nghiên cứu về các lỗ hổng bảo mật của trình đọc PDF là rất cần thiết để có thể phòng ngừa và khắc phục kịp thời.

Trong những năm gần đây, công nghệ thông tin (CNTT) đã và đang được ứng dụng vào rất nhiều ngành nghề và lĩnh vực khác nhau, góp phần to lớn vào việc thúc đẩy sự phát triển kinh tế. Tuy nhiên, sự phát triển này cũng kéo theo sự gia tăng các mối đe dọa bảo mật. Đặc biệt, trong lĩnh vực quản lý tài liệu số, việc bảo vệ các file PDF và hệ thống đọc PDF trở nên cực kỳ quan trọng.

Với tính thiết thực của đề tài, nhóm em đã chọn đề tài: “Tìm hiểu về trình đọc file PDF và khai thác các lỗ hổng của trình đọc file PDF (Client-Side)” với mục đích nghiên cứu, tìm hiểu và áp dụng các kỹ thuật để xác định các lỗ hổng bảo mật trong trình đọc PDF, đồng thời đề xuất các giải pháp để bảo vệ hệ thống khỏi các cuộc tấn công. Qua đó, chúng em hi vọng có thể đóng góp vào việc nâng cao nhận thức về bảo mật thông tin và đảm bảo an toàn cho người dùng.

* 1. Mục tiêu của đề tài

Nhằm hoàn tất chương trình học và tiếp thu thêm nhiều kiến thức về bảo mật và khai thác lỗ hổng, chúng em đã chọn đề tài này. Trên cơ sở nghiên cứu, tìm tòi qua các tài liệu khoa học và những kiến thức đã học vào thực tế để đánh giá, phân tích các lỗ hổng bảo mật trong trình đọc PDF, từ đó đưa ra các giải pháp nhằm nâng cao chất lượng bài làm của mình trở nên hoàn thiện hơn.

* 1. Đối tượng, phạm vi và phương pháp tiếp cận

**Đối tượng nghiên cứu:** Trình đọc file PDF và các lỗ hổng bảo mật (Client-Side).

**Cơ sở phương pháp nghiên cứu:**

* Nghiên cứu qua các tài liệu khoa học, sách và các bài báo về bảo mật thông tin.
* Tìm hiểu thêm các thông tin qua các nguồn tài liệu uy tín trên internet và thư viện nhà trường.
* Phương pháp truy cập internet có độ tin cậy cao để nghiên cứu đề tài.

**Phương tiện nghiên cứu:** Qua các trang sách, bài báo khoa học và các nguồn tài liệu trên internet.

**Mô tả công việc:**

* Tổng hợp các kết quả nghiên cứu từ các tư liệu liên quan.
* Sử dụng các kiến thức đã có và các công cụ để phát hiện và khai thác lỗ hổng bảo mật.
* Quan sát và phân tích các lỗ hổng bảo mật của các trình đọc PDF phổ biến.
* Góp ý và hướng dẫn từ giảng viên.
  1. Đóng góp của đề tài

Trên cơ sở tìm hiểu, phân tích, tổng hợp và áp dụng, đề tài đạt được một số kết quả như sau:

* Hoàn thành báo cáo phân tích và khai thác lỗ hổng bảo mật của trình đọc PDF đúng với các yêu cầu học thuật.
* Ứng dụng được các kiến thức đã học về bảo mật thông tin vào việc phân tích và khai thác lỗ hổng bảo mật của trình đọc PDF, có tính ứng dụng thực tiễn cao.
  1. Bố cục của đề tài

Sau phần *Mở đầu*, báo cáo được trình bày trong ba chương, cụ thể như sau:

Chương 1. *Tổng quan về trình đọc file PDF*. Trình bày các khái niệm và tổng quan về trình đọc file PDF, bao gồm lịch sử phát triển, các trình đọc PDF phổ biến, cấu trúc của file PDF, và các công cụ, môi trường sử dụng để phát triển và kiểm tra trình đọc PDF.

Chương 2. *Các lỗ hổng tồn tại và phương thức khai thác trên trình đọc PDF*

Nội dung chương bao gồm: Tổng quan về các lỗ hổng bảo mật phổ biến trong trình đọc PDF, phân loại, phân tích cụ thể và phương thức khai thác các lỗ hổng theo các tiêu chí, và các ví dụ cụ thể về các cuộc tấn công.

Chương 3. *Mô phỏng và phòng chống cách thức khai thác lỗ hổng*

Chương này trình bày các kết quả và giải pháp bảo mật, bao gồm:

Tiến hành mô phỏng cách thức khai thác lỗ hổng đã nêu trong Chương 2 và trình bày các biện pháp phòng chống được triển khai để khắc phục lỗ hổng đó. Nội dung bao gồm các bước mô phỏng chi tiết, kiểm tra hiệu quả của các biện pháp phòng chống và đánh giá kết quả.

Cuối cùng là *Kết luận*, *Tài liệu tham khảo* và *Phụ lục* liên quan đến đề tài.

# 

# TỔNG QUAN VỀ CÔNG CỤ VÀ MÔI TRƯỜNG PHÁT TRIỂN

## Khái niệm về PDF và trình đọc file PDF

### Giới thiệu về PDF

Nguồn gốc: Định dạng PDF được phát triển bởi Adobe Systems vào năm 1993 nhằm đáp ứng nhu cầu truyền tải tài liệu dưới một định dạng không phụ thuộc vào phần cứng, phần mềm hay hệ điều hành.



Hình 1.1.1: Biểu tượng của định dạng file PDF

Đặc điểm chính: PDF có khả năng lưu giữ bố cục tài liệu, bao gồm phông chữ, hình ảnh, đồ họa, và siêu dữ liệu. Điều này giúp nội dung được bảo toàn khi chia sẻ giữa các thiết bị khác nhau.

Mục tiêu sử dụng: Định dạng này thường được dùng trong tài liệu chính thức, hợp đồng, báo cáo, và tài liệu yêu cầu tính bảo mật và tính nguyên vẹn của định dạng.

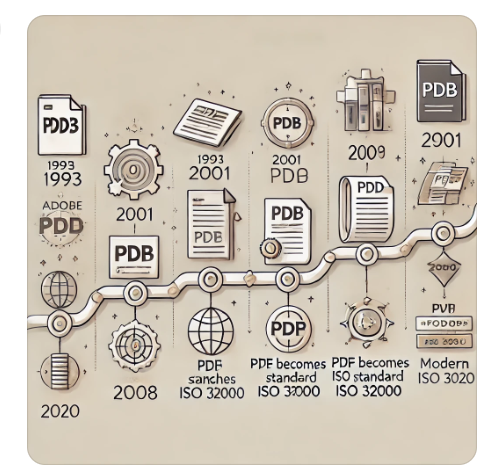
### Khái niệm trình đọc PDF

Chức năng cơ bản: Trình đọc PDF là phần mềm cho phép mở, xem và in các tài liệu PDF. Một số trình đọc còn cung cấp các công cụ ghi chú, chỉnh sửa và bảo mật.

Tính năng mở rộng: Một số trình đọc PDF hiện nay hỗ trợ các tính năng cao cấp hơn như điền biểu mẫu, chữ ký số, và mã hóa tài liệu. Một số còn có thể tích hợp các tính năng cho phép nhúng nội dung đa phương tiện và điều khiển các tập lệnh JavaScript trong PDF.

## Lịch sử phát triển của PDF và các trình đọc PDF

### Sự ra đời và phát triển của định dạng PDF



Hình 1.2.1: Quá trình phát triển của PDF qua các mốc thời gian

Giai đoạn khởi đầu: PDF được Adobe phát hành lần đầu vào năm 1993 trong bộ công cụ Adobe Acrobat, nhằm giải quyết vấn đề chia sẻ tài liệu đa nền tảng.

Mở rộng và chuẩn hóa: Định dạng PDF đã trở thành chuẩn mở ISO 32000 vào năm 2008, nhờ đó PDF trở thành một tiêu chuẩn quốc tế. Điều này đã mở rộng tính tương thích và ứng dụng của PDF, giúp PDF trở thành chuẩn phổ biến trong truyền thông tài liệu.

### Sự phát triển của các trình đọc PDF



Hình 1.2.2: Giao diện của các trình đọc PDF phổ biến

Adobe Acrobat Reader: Ra mắt cùng với định dạng PDF, Adobe Acrobat Reader là trình đọc đầu tiên và là một công cụ không thể thiếu cho PDF. Trình đọc này thường xuyên cập nhật và hiện vẫn là lựa chọn hàng đầu với nhiều tính năng bảo mật và chỉnh sửa nâng cao.

Các trình đọc khác: Sau sự thành công của Adobe Reader, các trình đọc PDF khác lần lượt ra đời như:

Foxit Reader: Nổi bật nhờ dung lượng nhẹ, hỗ trợ tốt cho doanh nghiệp.

Nitro PDF Reader: Có khả năng chuyển đổi PDF và tích hợp nhiều tính năng chỉnh sửa.

Trình đọc PDF tích hợp trên trình duyệt: Google Chrome, Firefox và Microsoft Edge đều tích hợp trình đọc PDF đơn giản cho phép xem tài liệu trực tiếp trên trình duyệt.

## Các trình đọc PDF phổ biến

### Adobe Acrobat Reader

Lịch sử và phổ biến: Đây là trình đọc PDF lâu đời và phổ biến nhất, với sự hỗ trợ liên tục từ Adobe Systems.

Tính năng nổi bật: Hỗ trợ các công cụ ghi chú, bảo mật, mã hóa và điền biểu mẫu. Adobe Acrobat Reader cũng có thể xử lý các tệp PDF có chứa mã JavaScript và các yếu tố tương tác.



Hình 1.3.1: Adobe Acrobat Reader

Các phiên bản: Phiên bản miễn phí chủ yếu dùng để đọc PDF, trong khi các phiên bản trả phí hỗ trợ nhiều tính năng nâng cao.

### Foxit Reader



Hình 1.3.2: Foxit Reader

Đặc điểm: Foxit Reader là một trình đọc nhẹ, hỗ trợ tốt cho doanh nghiệp và người dùng cá nhân.

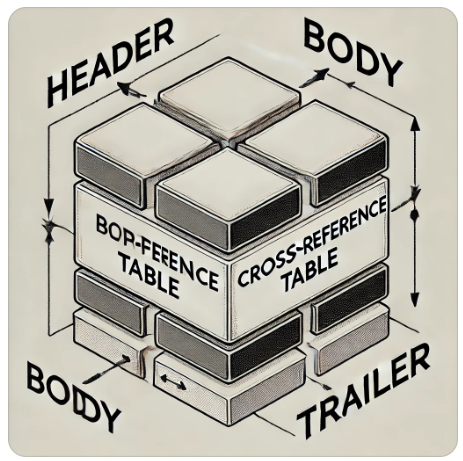
Các tính năng an toàn: Foxit Reader có các tính năng bảo mật như mã hóa, chống mã độc, và bảo vệ mật khẩu cho tài liệu PDF. Ngoài ra, trình đọc này cũng hỗ trợ tính năng chia sẻ và ghi chú tài liệu.

### Trình đọc PDF tích hợp trên trình duyệt

Google Chrome, Firefox và Microsoft Edge: Các trình duyệt hiện nay đều tích hợp trình đọc PDF, cho phép người dùng mở và xem các tài liệu PDF mà không cần cài đặt thêm phần mềm.

Hạn chế: Các trình đọc này chỉ hỗ trợ các chức năng cơ bản như xem và in, không có các tính năng chỉnh sửa hay bảo mật nâng cao.

## Cấu trúc của file PDF



Hình 1.4.1: Cấu trúc của file PDF

### Header

Chức năng: Header là phần đầu tiên của tệp PDF, chứa thông tin về phiên bản PDF mà tệp đang sử dụng (ví dụ: %PDF-1.4).

Tầm quan trọng: Các trình đọc dựa vào thông tin này để hiểu cách xử lý file PDF. Nếu có bất kỳ sự thay đổi nào trong định dạng Header, trình đọc có thể không mở được tệp.

### Body

Mô tả: Body là phần thân chính, chứa các đối tượng PDF như văn bản, hình ảnh, form và các yếu tố đa phương tiện khác.

Đối tượng trong Body: Các đối tượng này có thể được định nghĩa bằng các lệnh mã hóa và có thể bao gồm lệnh JavaScript hoặc các script khác, dễ bị khai thác nếu không được xử lý đúng cách.

### Cross-reference Table

Chức năng: Bảng Cross-reference lưu vị trí của các đối tượng PDF trong file, giúp trình đọc xác định các đối tượng để hiển thị chính xác.

Cấu trúc: Mỗi đối tượng được định vị bằng một số thứ tự trong bảng, có thể ảnh hưởng đến cách trình đọc PDF hiển thị hoặc xử lý nội dung.

### Trailer

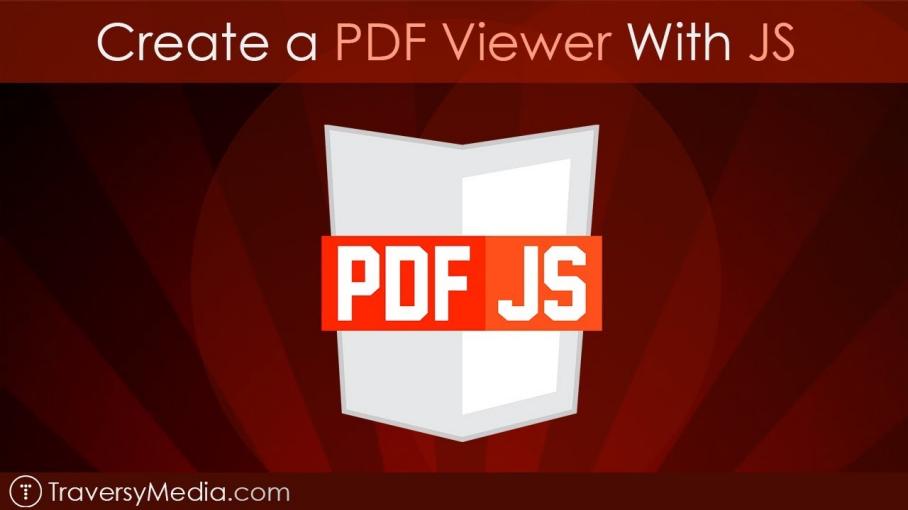
Mục đích: Trailer là phần cuối cùng của file, chứa thông tin cần thiết để trình đọc có thể truy cập nhanh vào các đối tượng quan trọng của file PDF.

Đặc tính bảo mật: Thông tin trong trailer ảnh hưởng đến khả năng kiểm soát và hiển thị nội dung file PDF, đôi khi có thể chứa các yếu tố bảo mật như thông tin mã hóa.

## Các công cụ và môi trường phát triển, kiểm tra trình đọc PDF

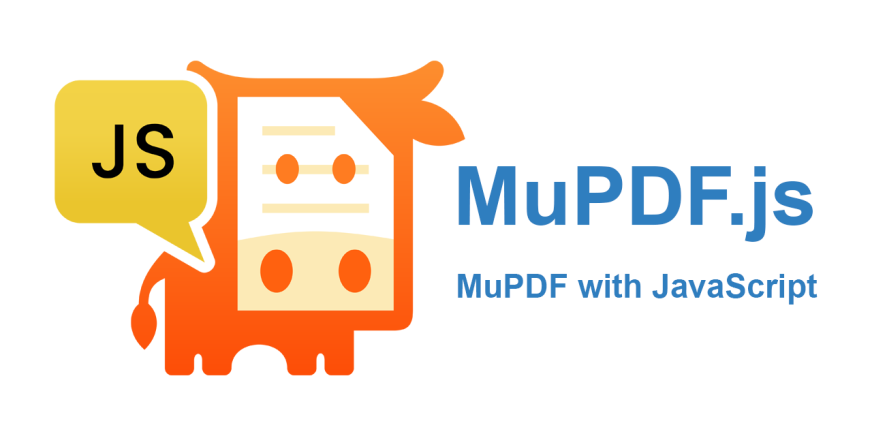
### Công cụ phát triển

PDF.js: Một thư viện mã nguồn mở được phát triển bởi Mozilla, cho phép hiển thị và xử lý PDF trên các trang web sử dụng JavaScript.



Hình 1.5.1: Công cụ PDF JS

MuPDF: Công cụ này mạnh mẽ và nhẹ, hỗ trợ tốt cho các ứng dụng yêu cầu khả năng xử lý PDF cao cấp.

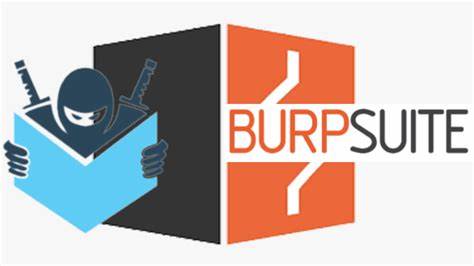


Hình 1.5.2: Công cụ MuPDF.

### Các công cụ kiểm tra bảo mật PDF

PDFiD và PDF-Parser: Đây là hai công cụ hữu ích cho phân tích cấu trúc và các yếu tố độc hại tiềm ẩn trong PDF.

Burp Suite và OWASP ZAP: Hai công cụ kiểm thử bảo mật phổ biến cho ứng dụng web, giúp phát hiện lỗ hổng bảo mật khi xử lý file PDF.



Hình 1.5.3: Burp Suite phân tích bảo mật PDF

Sandbox Environment: Sử dụng môi trường sandbox như Cuckoo Sandbox để phân tích các hành vi nguy hiểm tiềm ẩn của file PDF mà không ảnh hưởng đến hệ thống chính.

## Metasploit



Hình 1.6.1: Metasploit

Metasploit Framework là một nền tảng mã nguồn mở phổ biến dành cho việc kiểm tra thâm nhập (penetration testing), nghiên cứu bảo mật và phát triển khai thác lỗ hổng (exploit development). Được phát triển lần đầu vào năm 2003 bởi HD Moore, Metasploit hiện nay là một công cụ mạnh mẽ trong lĩnh vực an ninh mạng và được sử dụng rộng rãi bởi các chuyên gia bảo mật, nhà nghiên cứu và cả những kẻ tấn công.

## Tổng kết

Tóm tắt nội dung: Chương này cung cấp kiến thức cơ bản về định dạng PDF, các trình đọc phổ biến, cấu trúc và các công cụ hỗ trợ phát triển cũng như kiểm tra bảo mật.

Tầm quan trọng: Việc hiểu rõ các thành phần của PDF cũng như các lỗ hổng tiềm tàng là nền tảng quan trọng để thực hiện nghiên cứu và mô phỏng khai thác lỗ hổng trên các trình đọc PDF.

# Các lỗ hổng tồn tại và phương thức khai thác trên trình đọc PDF

## Tổng quan về lỗ hổng trên trình đọc PDF

### Định nghĩa và tầm quan trọng của bảo mật trong trình đọc PDF

**Lỗ hổng bảo mật** là các điểm yếu hoặc sai sót trong phần mềm mà tin tặc có thể lợi dụng để xâm nhập, kiểm soát hoặc gây ra thiệt hại cho hệ thống. Trong ngữ cảnh của trình đọc PDF, lỗ hổng bảo mật có thể dẫn đến các rủi ro nghiêm trọng như thực thi mã độc, mất dữ liệu, hoặc vi phạm quyền riêng tư.

**Tầm quan trọng của bảo mật trong trình đọc PDF** xuất phát từ việc PDF là một định dạng tài liệu phổ biến, được sử dụng rộng rãi trong các hoạt động kinh doanh, giáo dục, và cá nhân. Các tài liệu PDF có thể chứa thông tin nhạy cảm như báo cáo tài chính, hợp đồng, tài liệu y tế, và nhiều hơn nữa. Nếu bị tấn công, hậu quả có thể rất nghiêm trọng:

**Bảo vệ dữ liệu nhạy cảm:** Nhiều tài liệu PDF chứa thông tin quan trọng mà nếu bị lộ có thể gây thiệt hại nghiêm trọng cho cá nhân hoặc tổ chức.

**Ngăn chặn mã độc:** Các lỗ hổng có thể bị lợi dụng để chèn mã độc vào file PDF, từ đó lây nhiễm và chiếm quyền kiểm soát thiết bị của người dùng.

**Bảo vệ uy tín:** Một tổ chức hoặc cá nhân bị tấn công qua lỗ hổng PDF có thể mất uy tín và gặp phải nhiều vấn đề pháp lý.

### Các loại lỗ hổng phổ biến trên trình đọc PDF

Có nhiều loại lỗ hổng phổ biến trên các trình đọc PDF, mỗi loại có cách thức và mục tiêu khai thác riêng. Dưới đây là các loại lỗ hổng thường gặp:

**Lỗ hổng thực thi mã từ xa (Remote Code Execution - RCE):**

**Mô tả:** Đây là loại lỗ hổng nghiêm trọng cho phép kẻ tấn công thực thi mã độc trên hệ thống của nạn nhân mà không cần sự tương tác trực tiếp từ phía họ.

**Hậu quả:** Kẻ tấn công có thể kiểm soát toàn bộ hệ thống, đánh cắp thông tin, cài đặt phần mềm độc hại, và nhiều hơn nữa.

**Lỗ hổng XSS (Cross-Site Scripting):**

**Mô tả:** Lỗ hổng này cho phép kẻ tấn công chèn mã JavaScript độc hại vào tài liệu PDF, và khi người dùng mở tệp này, mã JavaScript sẽ được thực thi trên trình duyệt của họ.

**Hậu quả:** Kẻ tấn công có thể lấy cắp thông tin nhạy cảm như cookie, thông tin đăng nhập, hoặc thực hiện các hành động khác trên trình duyệt của người dùng.

**Lỗ hổng lừa đảo (Phishing):**



Hình 2.1.1: Tấn công phishing qua PDF

**Mô tả:** Tin tặc sử dụng tài liệu PDF để tạo ra các liên kết giả mạo hoặc giao diện lừa đảo nhằm lấy cắp thông tin cá nhân của người dùng.

**Hậu quả:** Người dùng có thể bị lừa nhập thông tin nhạy cảm như tên đăng nhập, mật khẩu, thông tin thẻ tín dụng vào các giao diện hoặc liên kết giả mạo.

**Lỗ hổng tiết lộ thông tin (Information Disclosure):**

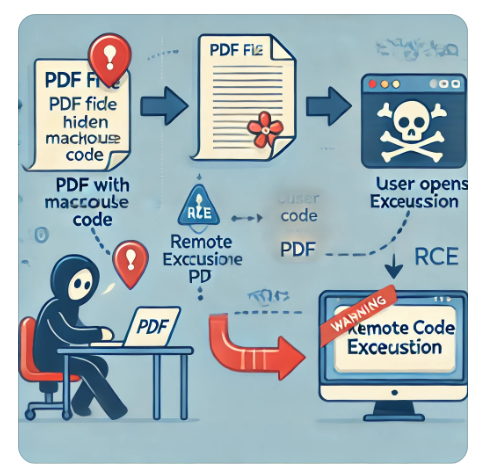
**Mô tả:** Các thông tin nhạy cảm có thể bị lộ qua metadata hoặc các phần dữ liệu ẩn trong tài liệu PDF do quản lý không đúng cách.

**Hậu quả:** Thông tin cá nhân, thông tin tổ chức hoặc các thông tin nhạy cảm khác có thể bị lộ và bị kẻ xấu lợi dụng.

## Lỗ hổng thực thi mã từ xa (Remote Code Execution - RCE)

### Cách thức hoạt động của lỗ hổng RCE

**Lỗ hổng thực thi mã từ xa (RCE)** cho phép kẻ tấn công chạy mã độc trên hệ thống của nạn nhân mà không cần sự tương tác trực tiếp. Cơ chế hoạt động của RCE chủ yếu dựa trên việc chèn mã độc vào file PDF. Khi người dùng mở tệp PDF này, mã độc sẽ được kích hoạt và thực thi, dẫn đến các hậu quả nghiêm trọng.



Hình 2.2.1: Cơ chế hoạt động của lỗ hổng thực thi mã từ xa trên file PDF

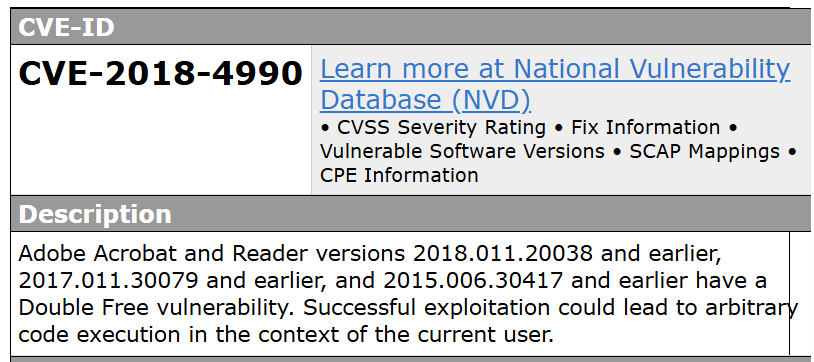
Quá trình khai thác RCE:

**Tạo file PDF độc hại**: Tin tặc tạo ra một tài liệu PDF chứa mã độc, thường bằng cách chèn mã khai thác vào các phần tử như JavaScript, biểu mẫu hoặc các đối tượng nhúng khác trong PDF.

**Phát tán file PDF độc hại**: File PDF độc hại được gửi đến nạn nhân qua email, tải về từ website, hoặc qua các phương thức khác.

**Thực thi mã độc**: Khi nạn nhân mở tệp PDF, mã độc sẽ được thực thi. Điều này có thể dẫn đến việc tải xuống và chạy mã độc bổ sung, đánh cắp thông tin, hoặc chiếm quyền điều khiển hệ thống.

Ví dụ về cơ chế khai thác RCE:



Hình 2.2.2: Lỗ hổng **CVE-2018-4990**

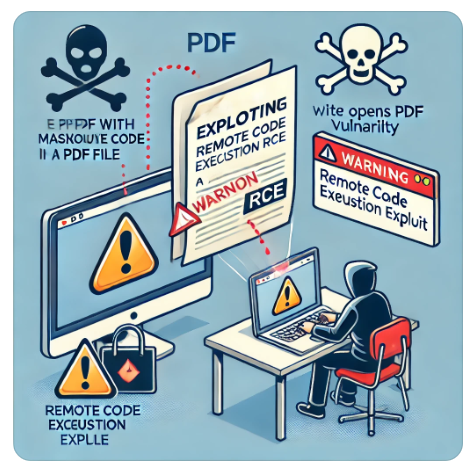
**Exploit CVE-2018-4990**: Đây là một lỗ hổng trong Adobe Acrobat và Reader cho phép thực thi mã từ xa khi xử lý các đối tượng nhúng trong PDF. Lỗ hổng này đã được sử dụng để tấn công người dùng qua các email lừa đảo chứa file PDF độc hại.

### Ví dụ về khai thác lỗ hổng RCE

Ví dụ 1: Tấn công qua email phishing

Một ví dụ điển hình về khai thác RCE là việc tin tặc gửi một email phishing với tập tin đính kèm PDF độc hại. Khi người dùng mở tệp, mã độc trong file PDF sẽ được thực thi, cho phép kẻ tấn công cài đặt phần mềm gián điệp hoặc kiểm soát toàn bộ hệ thống.

* **Chi tiết tấn công**: Email được thiết kế giống như thông báo từ một nguồn tin cậy (như ngân hàng, cơ quan chính phủ) yêu cầu người dùng mở file PDF để xem chi tiết quan trọng. File PDF chứa mã khai thác lỗ hổng RCE và mã độc được thực thi ngay khi file được mở.
* **Hậu quả**: Hệ thống của người dùng bị xâm nhập, thông tin cá nhân và dữ liệu nhạy cảm bị đánh cắp.



Hình 2.2.3: Ví dụ về khai thác lỗ hổng thực thi mã từ xa trên file PDF

Ví dụ 2: Tấn công qua tải xuống từ website

Tin tặc cũng có thể phát tán file PDF độc hại qua các website không an toàn. Khi người dùng tải xuống và mở file PDF từ một website lạ hoặc không tin cậy, mã độc sẽ được thực thi.

* **Chi tiết tấn công**: Người dùng truy cập vào một website không an toàn và tải xuống tài liệu PDF chứa mã khai thác RCE. Mã độc trong file PDF có thể tự động chạy và cài đặt phần mềm độc hại mà người dùng không hề hay biết.
* **Hậu quả**: Người dùng có thể mất quyền kiểm soát hệ thống, dữ liệu cá nhân bị lộ, và hệ thống có thể bị phần mềm độc hại chiếm đoạt.

Phương pháp phòng ngừa RCE:

**Cập nhật phần mềm**: Luôn cập nhật phiên bản mới nhất của trình đọc PDF để đảm bảo các lỗ hổng bảo mật đã được vá.

**Sử dụng phần mềm diệt virus**: Cài đặt và thường xuyên cập nhật phần mềm diệt virus để phát hiện và ngăn chặn các tệp PDF độc hại.

**Kiểm tra nguồn gốc của tệp PDF**: Chỉ mở các tệp PDF từ các nguồn tin cậy, và thận trọng với các email và liên kết không rõ nguồn gốc.

**Sử dụng các công cụ bảo mật bổ sung**: Sử dụng các công cụ bổ sung như sandboxing để kiểm tra tệp PDF trong một môi trường an toàn trước khi mở.

## Lỗ hổng XSS (Cross-Site Scripting) trên trình đọc PDF

### Cách thức hoạt động của lỗ hổng XSS trong PDF

**Lỗ hổng XSS (Cross-Site Scripting)** là một dạng tấn công trong đó kẻ tấn công chèn mã JavaScript độc hại vào tài liệu PDF. Khi nạn nhân mở tệp PDF, mã JavaScript này sẽ được thực thi trong ngữ cảnh của trình duyệt web, gây ra các hậu quả nghiêm trọng như đánh cắp thông tin cá nhân, chiếm quyền điều khiển tài khoản, hoặc thực hiện các hành động không mong muốn trên trang web.



Hình 2.3.1: Cơ chế của lỗ hổng XSS trên trình đọc PDF

Quá trình khai thác XSS trong PDF:

**Chèn mã độc vào PDF:** Kẻ tấn công tạo ra một tài liệu PDF với mã JavaScript độc hại được nhúng trong các phần tử như chú thích, biểu mẫu, hoặc các đoạn mã nhúng khác.

**Phát tán file PDF độc hại:** File PDF này được gửi tới nạn nhân qua email, tải về từ các trang web không an toàn, hoặc chia sẻ qua các phương tiện khác.

**Thực thi mã độc:** Khi nạn nhân mở file PDF trong trình đọc PDF hỗ trợ JavaScript, mã độc sẽ được thực thi, từ đó kẻ tấn công có thể thực hiện các hành động như đánh cắp cookie, chiếm quyền điều khiển tài khoản, hoặc tạo ra các yêu cầu giả mạo tới trang web khác.

Ví dụ về cơ chế khai thác XSS:

**Exploit XSS trong PDF:** Một tài liệu PDF chứa mã JavaScript độc hại được chèn vào các chú thích. Khi nạn nhân mở file PDF này, mã JavaScript sẽ chạy và gửi cookie của nạn nhân tới máy chủ của kẻ tấn công.

### Ví dụ về khai thác XSS trên PDF

Ví dụ 1: Tấn công qua email phishing

Kẻ tấn công gửi một email phishing với tệp đính kèm PDF chứa mã JavaScript độc hại. Khi người dùng mở file PDF này, mã JavaScript sẽ tự động chạy và gửi dữ liệu cookie của nạn nhân về máy chủ của kẻ tấn công.

* **Chi tiết tấn công:** Email giả mạo từ một nguồn tin cậy (như ngân hàng, cơ quan chính phủ) yêu cầu người dùng mở file PDF để xem thông tin quan trọng. File PDF này chứa mã JavaScript độc hại trong các chú thích hoặc biểu mẫu.
* **Hậu quả:** Kẻ tấn công có thể lấy cắp thông tin đăng nhập hoặc thông tin cá nhân khác từ nạn nhân.



Hình 2.3.2: Ví dụ khai thác lỗ hổng XSS trên PDF

Ví dụ 2: Tấn công qua trang web tải xuống

Kẻ tấn công đặt file PDF chứa mã JavaScript độc hại lên một trang web không an toàn. Khi người dùng tải xuống và mở file PDF này, mã JavaScript sẽ được thực thi và gửi dữ liệu về máy chủ của kẻ tấn công.

* **Chi tiết tấn công:** Người dùng truy cập vào một trang web không an toàn và tải xuống file PDF chứa mã JavaScript độc hại. Khi mở file, mã này sẽ chạy và thực hiện các hành động mà kẻ tấn công mong muốn.
* **Hậu quả:** Người dùng có thể bị chiếm quyền điều khiển tài khoản, mất dữ liệu cá nhân, hoặc bị thực hiện các hành động giả mạo trên các trang web khác.

Phương pháp phòng ngừa XSS:

**Cập nhật phần mềm:** Luôn cập nhật phiên bản mới nhất của trình đọc PDF để đảm bảo các lỗ hổng bảo mật đã được vá.

**Tắt JavaScript trong PDF:** Nếu không cần thiết, tắt tính năng chạy JavaScript trong trình đọc PDF để ngăn chặn mã độc thực thi.

**Sử dụng phần mềm diệt virus:** Cài đặt và thường xuyên cập nhật phần mềm diệt virus để phát hiện và ngăn chặn các file PDF độc hại.

**Kiểm tra nguồn gốc của tệp PDF:** Chỉ mở các tệp PDF từ các nguồn tin cậy, và thận trọng với các email và liên kết không rõ nguồn gốc.

**Sử dụng các công cụ bảo mật bổ sung:** Sử dụng các công cụ bổ sung như sandboxing để kiểm tra tệp PDF trong một môi trường an toàn trước khi mở.

## Lỗ hổng lừa đảo (Phishing) qua PDF

### Phương thức hoạt động của lỗ hổng lừa đảo qua PDF

**Lỗ hổng lừa đảo qua PDF (Phishing)** là một kỹ thuật tấn công mà kẻ xấu sử dụng để lừa người dùng cung cấp thông tin nhạy cảm như mật khẩu, thông tin thẻ tín dụng, hoặc các thông tin cá nhân khác thông qua các tài liệu PDF giả mạo. Phương thức này thường lợi dụng sự tin tưởng của người dùng vào các tài liệu được gửi từ những nguồn được cho là tin cậy.



Hình 2.4.1: Phương thức lừa đảo qua file PDF

Quá trình khai thác lỗ hổng lừa đảo qua PDF:

**Tạo file PDF giả mạo:** Kẻ tấn công tạo ra các tệp PDF trông giống như các tài liệu chính thức từ các tổ chức như ngân hàng, cơ quan chính phủ, hoặc các dịch vụ trực tuyến phổ biến. Các tệp này có thể chứa các biểu mẫu, liên kết hoặc giao diện giả mạo yêu cầu người dùng nhập thông tin cá nhân.

**Phát tán file PDF lừa đảo:** Các tệp PDF này được gửi đến nạn nhân qua email, hoặc chia sẻ qua mạng xã hội và các phương tiện truyền thông khác.

**Lừa người dùng cung cấp thông tin:** Khi người dùng mở tệp PDF và tương tác với các biểu mẫu hoặc liên kết giả mạo, thông tin cá nhân của họ sẽ bị gửi tới kẻ tấn công thay vì trang web hợp pháp.

Ví dụ về cơ chế lừa đảo qua PDF:

**Email giả mạo ngân hàng:** Kẻ tấn công gửi một email giả mạo từ ngân hàng, yêu cầu người dùng mở tệp PDF để kiểm tra tài khoản của họ. Tệp PDF chứa một biểu mẫu giả mạo yêu cầu người dùng nhập thông tin đăng nhập và mã PIN.

### Ví dụ về lừa đảo qua PDF

Ví dụ 1: Lừa đảo qua email giả mạo từ tổ chức tài chính

Tin tặc gửi một email giả mạo từ một tổ chức tài chính yêu cầu người dùng mở tệp PDF để xác minh thông tin tài khoản. Tệp PDF này chứa giao diện đăng nhập giả mạo, khi người dùng nhập thông tin, dữ liệu sẽ bị gửi đến kẻ tấn công.

* **Chi tiết tấn công:** Email được thiết kế giống như từ ngân hàng, yêu cầu người dùng mở tệp PDF và nhập thông tin tài khoản để xác nhận giao dịch. Tệp PDF chứa một biểu mẫu giả mạo ghi nhận các thông tin mà người dùng nhập.
* **Hậu quả:** Thông tin đăng nhập ngân hàng của người dùng bị đánh cắp và sử dụng để chiếm đoạt tài khoản.

Ví dụ 2: Lừa đảo qua tệp PDF chứa liên kết giả mạo

Kẻ tấn công tạo ra một tệp PDF chứa các liên kết giả mạo dẫn đến các trang web lừa đảo. Khi người dùng nhấp vào các liên kết này, họ sẽ được dẫn đến một trang web giả mạo yêu cầu nhập thông tin cá nhân.

* **Chi tiết tấn công:** Tệp PDF được thiết kế như một tài liệu hướng dẫn từ một tổ chức tin cậy, chứa các liên kết đến các trang web giả mạo. Người dùng tin tưởng và nhấp vào các liên kết này, sau đó nhập thông tin cá nhân vào các trang giả mạo.
* **Hậu quả:** Thông tin cá nhân và dữ liệu nhạy cảm của người dùng bị đánh cắp và sử dụng cho các mục đích xấu.

Phương pháp phòng ngừa lừa đảo qua PDF:

**Kiểm tra nguồn gốc email:** Luôn kiểm tra kỹ nguồn gốc của email và các tệp đính kèm trước khi mở. Đặc biệt cẩn thận với các email yêu cầu thông tin nhạy cảm.

**Xác minh liên kết trước khi nhấp:** Trước khi nhấp vào bất kỳ liên kết nào trong tệp PDF, hãy kiểm tra xem liên kết đó có dẫn đến trang web chính thức hay không.

**Sử dụng phần mềm diệt virus:** Cài đặt và thường xuyên cập nhật phần mềm diệt virus để phát hiện và ngăn chặn các tệp PDF độc hại.

**Đào tạo người dùng:** Tăng cường nhận thức về các phương thức lừa đảo qua email và tệp PDF cho người dùng, giúp họ nhận diện và phòng tránh các cuộc tấn công.

**Sử dụng xác thực hai yếu tố (2FA):** Sử dụng phương pháp xác thực hai yếu tố cho các tài khoản quan trọng để tăng cường bảo mật.

## Lỗ hổng tiết lộ thông tin (Information Disclosure)

### Cách thức hoạt động của lỗ hổng tiết lộ thông tin

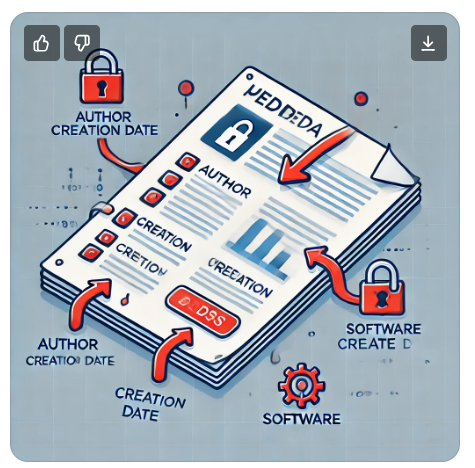
**Lỗ hổng tiết lộ thông tin (Information Disclosure)** xảy ra khi các thông tin nhạy cảm không được bảo vệ đúng cách và có thể bị lộ ra ngoài, đôi khi do sự thiếu sót trong việc xử lý dữ liệu hoặc các lỗi trong phần mềm. Điều này có thể bao gồm các thông tin như metadata của tài liệu, các phần dữ liệu ẩn trong file PDF, hoặc các thông tin không được mã hóa đúng cách.

Quá trình khai thác lỗ hổng tiết lộ thông tin:

**Xem metadata của file PDF:** Metadata là thông tin ẩn chứa trong các tệp PDF như tác giả, ngày tạo, phần mềm sử dụng để tạo file, và có thể bao gồm thông tin nhạy cảm nếu không được xóa bỏ đúng cách. Kẻ tấn công có thể dễ dàng truy cập và xem các thông tin này thông qua các công cụ phân tích PDF.

**Truy xuất dữ liệu ẩn trong file PDF:** Đôi khi, các tài liệu PDF có thể chứa các dữ liệu ẩn hoặc bị xóa không đúng cách, dẫn đến việc thông tin vẫn có thể được truy cập. Điều này bao gồm các phần tử như chú thích, nhận xét hoặc các bản nháp không được xóa bỏ hoàn toàn.

**Khai thác thông tin không mã hóa:** Nếu tài liệu PDF chứa thông tin nhạy cảm mà không được mã hóa, kẻ tấn công có thể dễ dàng truy cập và đọc các thông tin này khi có được tệp PDF.



Hình 2.5.1: Lỗ hổng tiết lộ thông tin trong metadata của file PDF

Ví dụ về cơ chế tiết lộ thông tin:

**Metadata lộ thông tin nhạy cảm:** Một tài liệu PDF được tạo bởi một tổ chức chứa metadata với thông tin về các nhân viên hoặc các dự án đang thực hiện, và thông tin này có thể bị lợi dụng nếu rơi vào tay kẻ xấu.

**Dữ liệu ẩn trong file PDF:** Một tài liệu PDF chứa các chú thích hoặc nhận xét về các thông tin nhạy cảm không được xóa bỏ trước khi chia sẻ, dẫn đến việc thông tin này bị lộ ra ngoài.

### Ví dụ về khai thác lỗ hổng tiết lộ thông tin

Ví dụ 1: Metadata lộ thông tin dự án

Một ví dụ về lỗ hổng tiết lộ thông tin là khi một tài liệu PDF được chia sẻ công khai mà không xóa bỏ metadata. Thông tin này có thể bao gồm tên của người tạo, ngày tạo, và các chi tiết liên quan đến dự án mà tài liệu đó đề cập.

* **Chi tiết tấn công:** Một tổ chức chia sẻ tài liệu PDF công khai mà không xóa bỏ metadata. Kẻ tấn công có thể sử dụng các công cụ phân tích PDF để trích xuất thông tin này, như tên của nhân viên phụ trách dự án, ngày hoàn thành, và các chi tiết quan trọng khác.
* **Hậu quả:** Thông tin nhạy cảm về dự án bị lộ, có thể dẫn đến rủi ro về bảo mật và cạnh tranh.

Ví dụ 2: Dữ liệu ẩn trong tài liệu PDF

Một ví dụ khác về lỗ hổng tiết lộ thông tin là khi một tài liệu PDF chứa các nhận xét hoặc chú thích nhạy cảm không được xóa bỏ hoàn toàn. Khi tài liệu được chia sẻ, những thông tin này có thể bị trích xuất bởi kẻ tấn công.

* **Chi tiết tấn công:** Một tài liệu PDF chứa các chú thích về thông tin tài chính nhạy cảm không được xóa bỏ trước khi chia sẻ. Khi tài liệu được gửi đến đối tác, họ có thể dễ dàng xem các chú thích này bằng cách sử dụng các công cụ phân tích PDF.
* **Hậu quả:** Thông tin tài chính nhạy cảm bị lộ ra ngoài, gây thiệt hại về uy tín và tài chính cho tổ chức.

Phương pháp phòng ngừa lỗ hổng tiết lộ thông tin:

**Xóa metadata trước khi chia sẻ:** Sử dụng các công cụ hoặc phần mềm để xóa bỏ metadata khỏi tài liệu PDF trước khi chia sẻ.

**Kiểm tra và loại bỏ dữ liệu ẩn:** Trước khi chia sẻ, kiểm tra và loại bỏ các phần dữ liệu ẩn hoặc các nhận xét, chú thích không cần thiết trong tài liệu PDF.

**Mã hóa tài liệu PDF:** Sử dụng các công cụ mã hóa để bảo vệ thông tin nhạy cảm trong tài liệu PDF. Điều này giúp ngăn chặn truy cập trái phép vào các thông tin quan trọng.

**Sử dụng phần mềm bảo mật chuyên dụng:** Các phần mềm bảo mật chuyên dụng có thể giúp kiểm tra và đảm bảo rằng tài liệu PDF không chứa các thông tin nhạy cảm trước khi chia sẻ.

**Đào tạo người dùng:** Tăng cường nhận thức về các rủi ro bảo mật liên quan đến tài liệu PDF cho người dùng, giúp họ hiểu rõ cách bảo vệ thông tin nhạy cảm.

## Các công cụ và phương pháp kiểm tra lỗ hổng trên trình đọc PDF

### Công cụ phát hiện và kiểm tra lỗ hổng



Hình 2.6.1: Các công cụ phân tích và kiểm tra lỗ hổng trên PDF

Trong quá trình kiểm tra và bảo mật cho trình đọc PDF, việc sử dụng các công cụ chuyên dụng là cần thiết để phát hiện và khắc phục các lỗ hổng bảo mật. Dưới đây là một số công cụ phổ biến được sử dụng trong lĩnh vực này:

**Burp Suite**

**Mô tả:** Burp Suite là một công cụ kiểm thử xâm nhập toàn diện được sử dụng rộng rãi để kiểm tra bảo mật web, bao gồm các tệp PDF. Công cụ này cung cấp các tính năng như quét lỗ hổng, kiểm tra các điểm yếu bảo mật và thử nghiệm các kịch bản tấn công.

**Tính năng:** Burp Suite có thể kiểm tra các lỗ hổng như SQL Injection, Cross-Site Scripting (XSS), và các điểm yếu khác trong tệp PDF và ứng dụng web.

**Cuckoo Sandbox**

**Mô tả:** Cuckoo Sandbox là một hệ thống phân tích mã độc tự động, có khả năng phân tích các tệp PDF để phát hiện các hành vi đáng ngờ và mã độc. Công cụ này cho phép kiểm tra và đánh giá các mối đe dọa tiềm ẩn trong tệp PDF.

**Tính năng:** Cuckoo Sandbox có thể phát hiện mã độc, phân tích hành vi của tệp PDF, và cung cấp báo cáo chi tiết về các hoạt động đáng ngờ.

**PDF-Analyzer**

**Mô tả:** PDF-Analyzer là một công cụ chuyên dụng để phân tích và kiểm tra các lỗ hổng trong tệp PDF. Công cụ này có khả năng kiểm tra các thành phần của tệp PDF, phát hiện các điểm yếu bảo mật và các mã độc tiềm ẩn.

**Tính năng:** PDF-Analyzer có thể kiểm tra các yếu tố như metadata, các đối tượng nhúng, và các thành phần khác trong tệp PDF để phát hiện lỗ hổng.

### Phương pháp kiểm tra lỗ hổng bảo mật trên trình đọc PDF

Kiểm tra lỗ hổng bảo mật trên trình đọc PDF đòi hỏi một quy trình chặt chẽ và khoa học để đảm bảo tất cả các điểm yếu tiềm ẩn được phát hiện và khắc phục. Dưới đây là các phương pháp kiểm tra lỗ hổng bảo mật phổ biến:

**Phân tích mã nguồn**

**Mô tả:** Phân tích mã nguồn của trình đọc PDF để tìm kiếm các lỗ hổng bảo mật tiềm ẩn trong mã. Điều này có thể bao gồm kiểm tra các đoạn mã xử lý tệp PDF, quản lý bộ nhớ, và các thành phần khác liên quan đến bảo mật.

**Kỹ thuật:** Sử dụng các công cụ phân tích tĩnh (static analysis) để kiểm tra mã nguồn, tìm kiếm các lỗi lập trình và các điểm yếu bảo mật. Các công cụ như SonarQube, Fortify Static Code Analyzer có thể hỗ trợ trong quá trình này.

**Kiểm tra metadata**

**Mô tả:** Kiểm tra metadata của các tệp PDF để đảm bảo không có thông tin nhạy cảm hoặc không cần thiết bị lộ ra ngoài. Metadata có thể chứa thông tin về tác giả, ngày tạo, phần mềm tạo file, và các chi tiết khác.

**Kỹ thuật:** Sử dụng các công cụ như ExifTool để xem và chỉnh sửa metadata của các tệp PDF, đảm bảo rằng tất cả các thông tin nhạy cảm đã được xóa bỏ.

**Thử nghiệm với công cụ bảo mật**

**Mô tả:** Sử dụng các công cụ bảo mật như Burp Suite, Cuckoo Sandbox và PDF-Analyzer để kiểm tra và phát hiện các lỗ hổng bảo mật trong tệp PDF. Các công cụ này có thể thực hiện các thử nghiệm tự động và cung cấp báo cáo chi tiết về các lỗ hổng.

**Kỹ thuật:** Thực hiện các thử nghiệm bảo mật bằng cách chạy các công cụ phân tích trên tệp PDF, kiểm tra các yếu tố như mã nhúng, các đoạn mã JavaScript, và các thành phần khác có thể chứa lỗ hổng.



Hình 2.6.2: Quy trình kiểm tra bảo mật cho file PDF

**Kiểm tra tính năng bảo mật**

**Mô tả:** Kiểm tra các tính năng bảo mật được cung cấp bởi trình đọc PDF, bao gồm mã hóa, bảo vệ bằng mật khẩu, và các cơ chế bảo vệ khác. Đảm bảo rằng các tính năng này được cấu hình đúng và hoạt động hiệu quả.

**Kỹ thuật:** Sử dụng các tệp PDF với các tính năng bảo mật được bật và thử nghiệm các kịch bản tấn công để đảm bảo rằng các tính năng này không thể bị bypass dễ dàng.

## Tổng kết

Bảo mật trong trình đọc PDF là cực kỳ quan trọng để bảo vệ dữ liệu và thông tin nhạy cảm khỏi các cuộc tấn công. Các lỗ hổng như thực thi mã từ xa (RCE), Cross-Site Scripting (XSS), lừa đảo (Phishing) và tiết lộ thông tin đều là những mối đe dọa nghiêm trọng. Để bảo vệ các tệp PDF, chúng ta cần:

**Cập nhật phần mềm**: Đảm bảo sử dụng phiên bản mới nhất của trình đọc PDF.

**Sử dụng phần mềm diệt virus**: Ngăn chặn các tệp PDF độc hại.

**Xóa metadata và dữ liệu ẩn**: Trước khi chia sẻ tài liệu.

**Kiểm tra nguồn gốc của tệp PDF**: Chỉ mở các tệp từ nguồn tin cậy.

**Sử dụng mã hóa**: Bảo vệ thông tin nhạy cảm.

Việc hiểu rõ và áp dụng các biện pháp bảo mật này sẽ giúp giảm thiểu các rủi ro và bảo vệ hệ thống khỏi các lỗ hổng bảo mật.



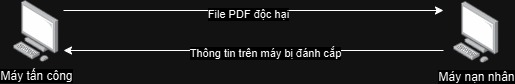
Hình 2.7.1: Tổng kết các lỗ hổng bảo mật trên trình đọc PDF

### Kết luận

Bảo mật tài liệu PDF và trình đọc PDF là một quá trình liên tục và cần được cập nhật thường xuyên để đối phó với các mối đe dọa mới. Việc hiểu rõ và áp dụng các biện pháp bảo mật sẽ giúp bảo vệ dữ liệu, đảm bảo quyền riêng tư và ngăn chặn các cuộc tấn công tiềm ẩn. Bằng cách thực hiện các biện pháp phòng ngừa và sử dụng các công cụ kiểm tra bảo mật hiệu quả, chúng ta có thể giảm thiểu các rủi ro và bảo vệ hệ thống khỏi các lỗ hổng bảo mật.

# MÔ PHỎNG

## Mô hình hệ thống



Hình 3.1.1: Sơ đồ hệ thống

## Chuẩn bị môi trường

- Chuẩn bị các máy ảo: Kali linux, Windows 7.

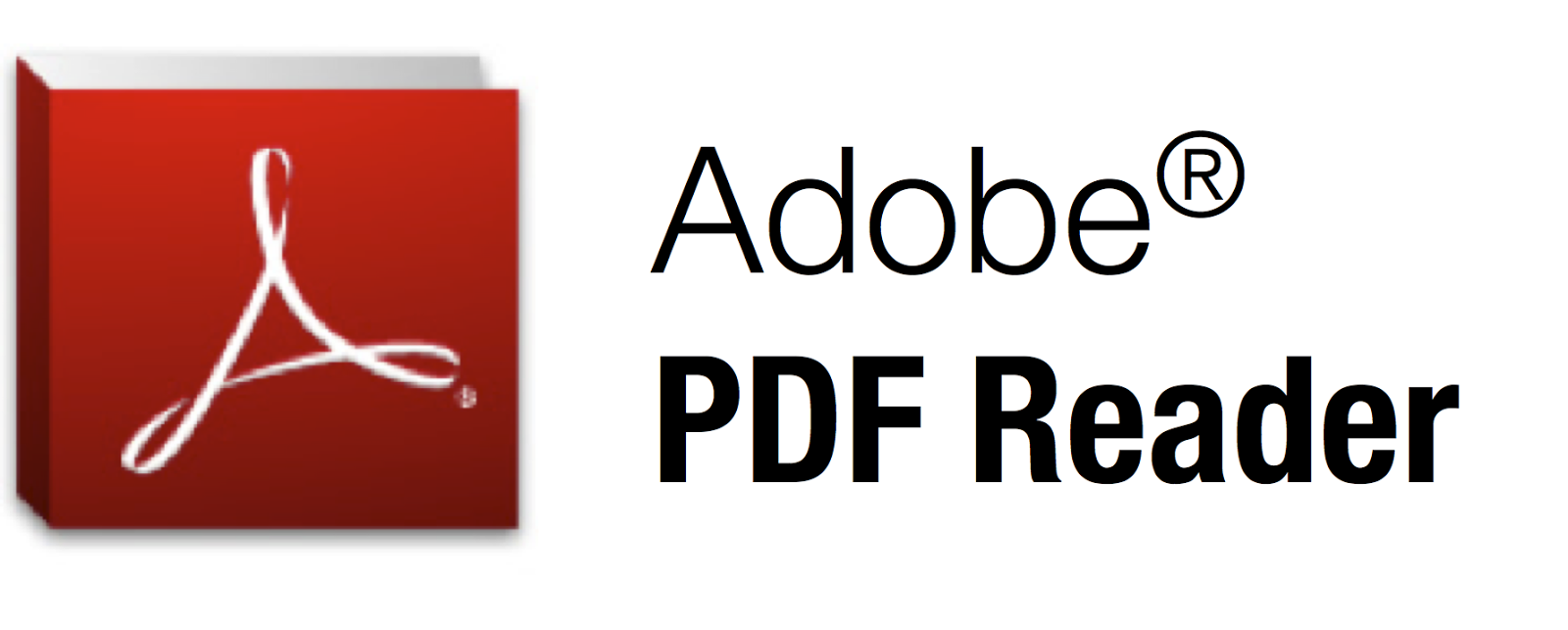


Hình 3.2.1: Kali Linux



Hình 3.2.2: Windows 7

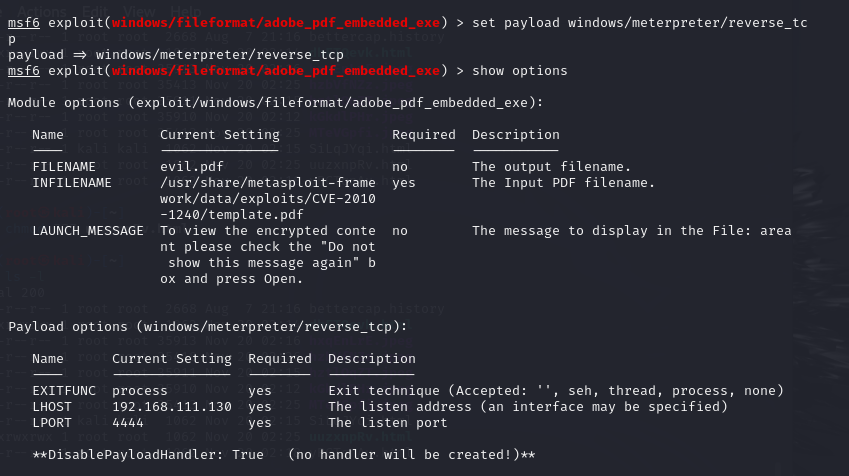
- Adobe reader 9.



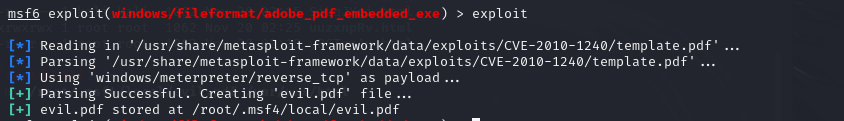
Hình 3.2.3: Adobe reader 9

## Mô phỏng lỗ hỏng Remote Code Execution

- Tạo file PDF độc hại



Hình 3.3.1: cấu hình tạo file PDF độc hại



Hình 3.3.2: Tạo thành công file PDF độc hại

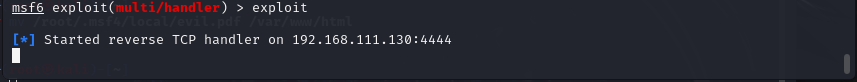
- Cấu hình lắng nghe kết nối trên máy tấn công



Hình 3.3.3: Thiết lập listener.



Hình 3.3.4: Cấu hình các thông số cho listener

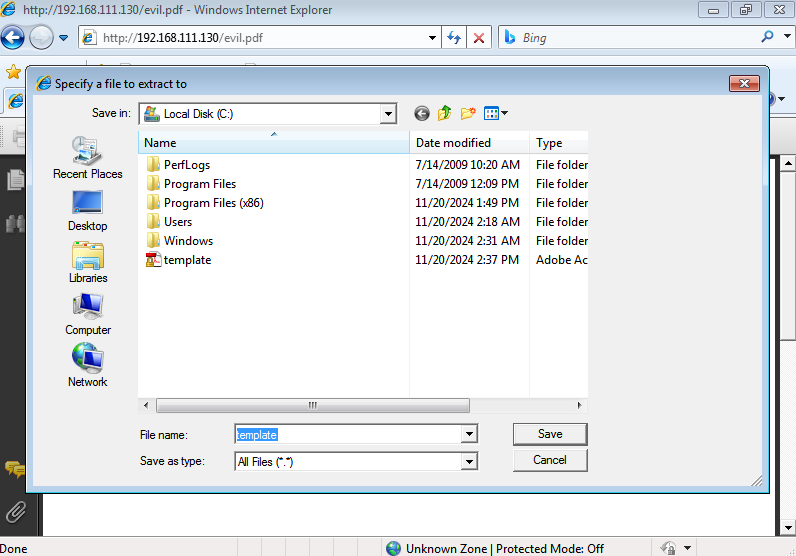


Hình 3.3.5: Chạy listener

- Phát tán file

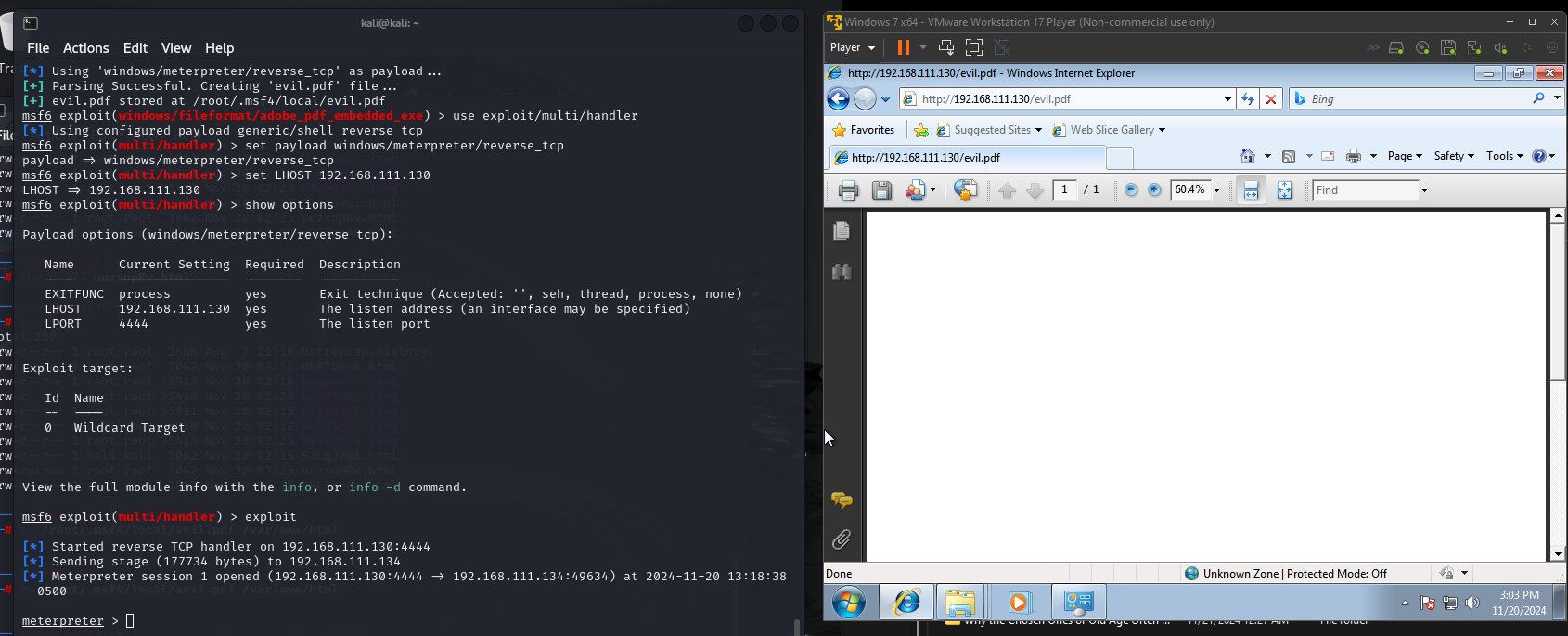


Hình 3.3.6: phát tán file qua web



Hình 3.3.7: Máy nạn nhân tải file độc hại về

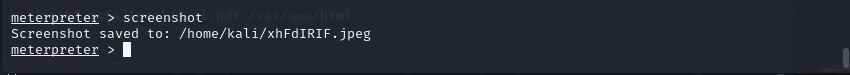
- Máy nạn nhân mở file thì bên máy tấn công sẽ nhận được kết nối



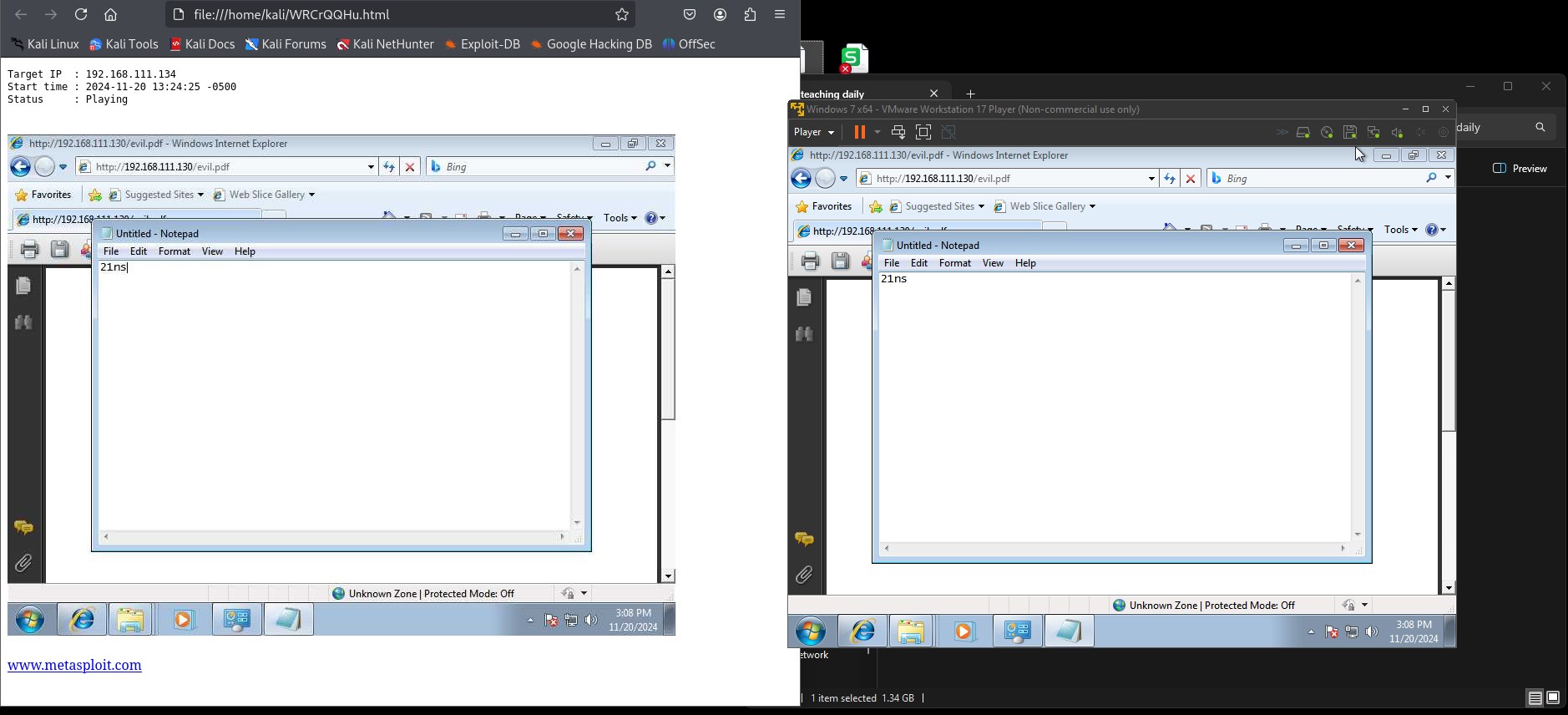
Hình 3.3.8: Máy tấn công đã kết nối được với máy nạn nhân

- Khai thác các thông tin:

+ Tiến hành thực hiện các hành động thu thập thông tin của nạn nhân



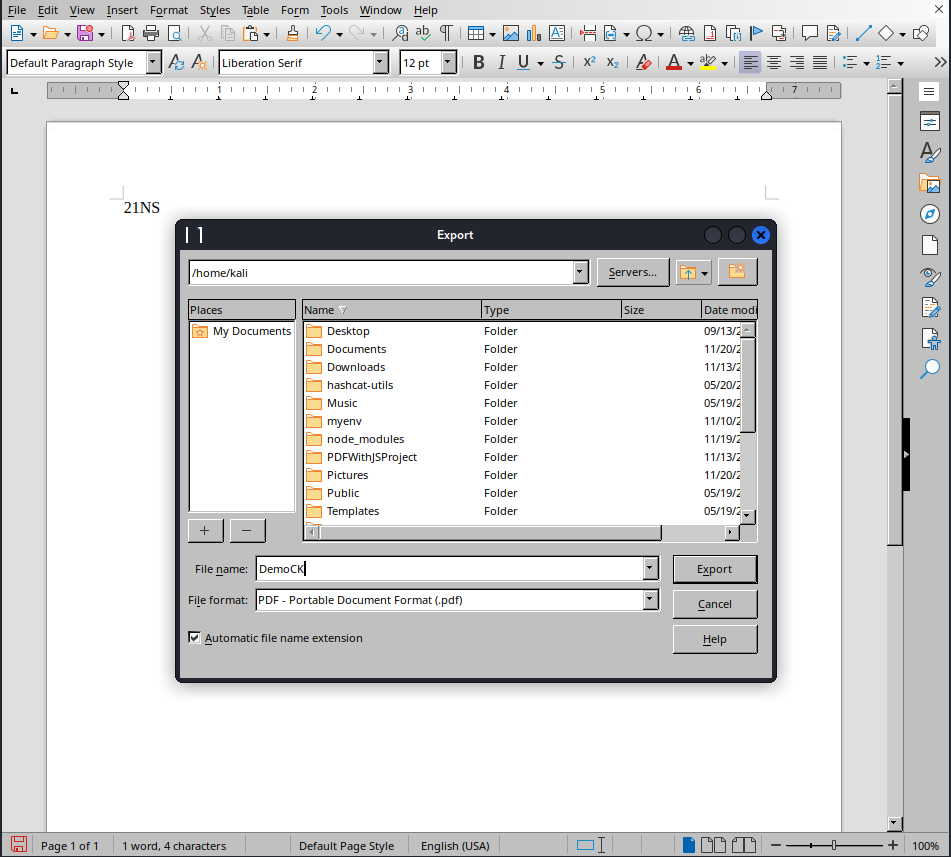
Hình 3.3.9: Chụp màn hình máy nạn nhân



Hình 3.3.10 Theo dõi hành động trực tiếp trên máy nạn nhân

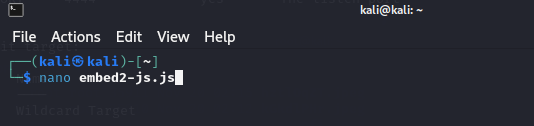
## Mô phỏng lỗ hỏng XSS

- Tạo file PDF đơn giản

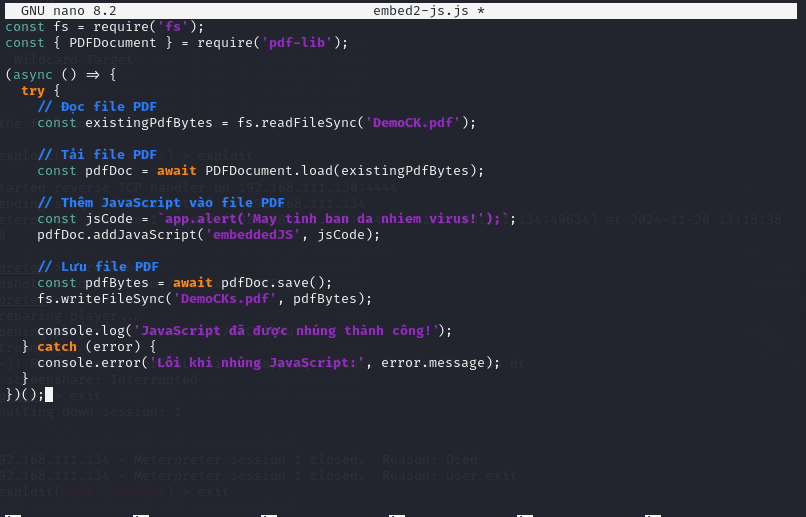


Hình 3.4.1: Tạo file PDF

- Tạo script Node.js để nhúng JavaScript

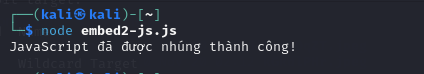


Hình 3.4.2: Tạo file JavaScipt



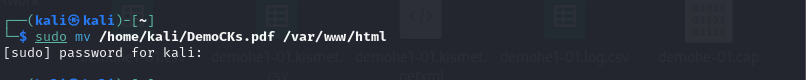
Hình 3.4.3: Nội dung file JS

- Nhúng JavaScript vào file PDF.



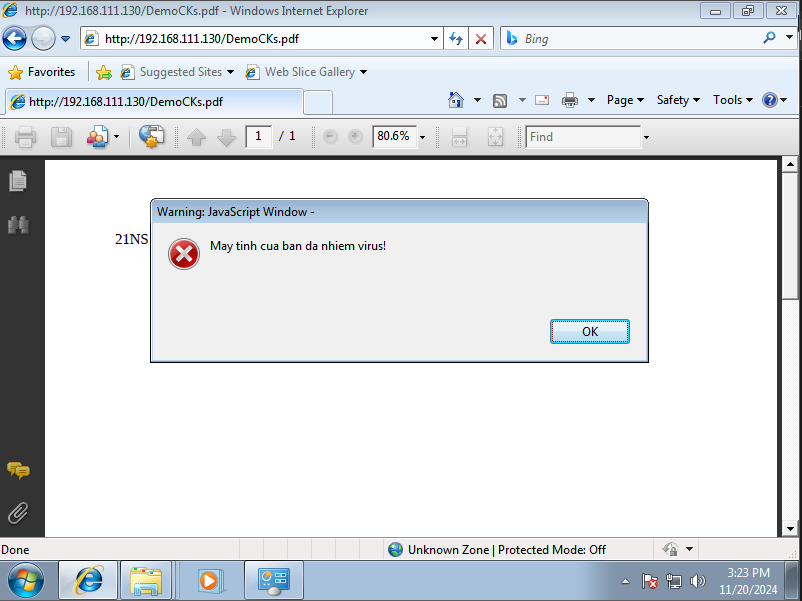
Hình 3.4.4: File PDF đã nhúng JavaScript thành công

- Phát tán file PDF.



Hình 3.4.5: Phát tán file độc hại qua Web

- Máy nạn nhân mở file và JSược thực hiện



Hình 3.4.6: File chứa JS đã được thực thi

KẾT LUẬN

* 1. Kết quả đạt được

Qua quá trình thực hiện đồ án “Tìm hiểu về trình đọc file PDF và khai thác các lỗ hổng của trình đọc file PDF (Client-side)”, em xin rút ra một số kết luận như sau:

**THUẬN LỢI**

**Kiến thức chuyên môn:** Được trang bị kiến thức vững vàng về bảo mật và các kỹ thuật tấn công, giúp dễ dàng tiếp cận và hiểu rõ vấn đề.

**Hỗ trợ từ giảng viên và đồng nghiệp:** Nhận được sự hỗ trợ và hướng dẫn tận tình từ giảng viên và các bạn đồng nghiệp, giúp giải quyết các khó khăn gặp phải trong quá trình nghiên cứu.

**Tài liệu tham khảo phong phú:** Có sẵn nhiều tài liệu tham khảo, bao gồm cả các nghiên cứu trước đây, bài báo khoa học và các công cụ kiểm tra bảo mật chuyên dụng.

**KHÓ KHĂN**

**Khó khăn về công cụ:** Một số công cụ phân tích và kiểm tra lỗ hổng có chi phí cao hoặc yêu cầu cấu hình phức tạp, gây khó khăn trong việc tiếp cận và sử dụng.

**Thời gian hạn chế:** Khối lượng công việc lớn và thời gian có hạn khiến cho việc nghiên cứu và thực nghiệm gặp nhiều thách thức.

**Cập nhật kiến thức:** Lĩnh vực bảo mật luôn thay đổi và phát triển, yêu cầu việc cập nhật kiến thức liên tục để theo kịp các xu hướng và kỹ thuật mới.

* 1. Những mặt đạt được và chưa đạt được

**Mặt đạt được:**

**Hiểu rõ các lỗ hổng bảo mật:** Nắm vững các loại lỗ hổng bảo mật phổ biến trên trình đọc PDF và cách thức khai thác chúng.

**Phát hiện và khắc phục:** Biết cách sử dụng các công cụ để phát hiện và khắc phục các lỗ hổng bảo mật trên trình đọc PDF.

**Đề xuất biện pháp bảo mật:** Đưa ra các biện pháp phòng ngừa và bảo vệ hiệu quả để giảm thiểu rủi ro bảo mật.

**Mặt chưa đạt:**

**Thực nghiệm chưa toàn diện:** Do thời gian và tài nguyên hạn chế, một số thử nghiệm chưa được thực hiện đầy đủ hoặc chưa có điều kiện kiểm chứng toàn diện.

**Thiếu cập nhật mới nhất:** Một số thông tin và phương pháp có thể đã lỗi thời và cần được cập nhật thêm các kỹ thuật và công cụ mới nhất.

3. Hướng nghiên cứu

Để tiếp tục hoàn thiện và phát triển nghiên cứu, em đề xuất một số hướng nghiên cứu tiếp theo:

**Nâng cao kiến thức:** Tiếp tục cập nhật và học hỏi các kiến thức mới về bảo mật và các kỹ thuật tấn công.

**Thực nghiệm toàn diện:** Thực hiện các thử nghiệm toàn diện hơn với các công cụ và môi trường khác nhau để kiểm chứng và đánh giá hiệu quả bảo mật.

**Phát triển công cụ kiểm tra:** Nghiên cứu và phát triển các công cụ kiểm tra bảo mật tự động, giúp nâng cao hiệu quả và tiết kiệm thời gian kiểm tra.

TÀI LIỆU THAM KHẢO

[1] Adobe Systems Incorporated. (2020). “Adobe Acrobat Reader Security Guide”. Adobe Systems Incorporated.

[2] Acunetix. (2021). "XSS Exploits in PDFs". Acunetix Blog.

[3] Mitre Corporation. (2018). "CVE-2018-4990: Remote Code Execution in Adobe Acrobat Reader".

[4] Kaspersky. (2020). "Preventing RCE Attacks". Kaspersky Resource Center.

[5] Burp Suite. (2021). "Burp Suite Documentation". PortSwigger Ltd.

[6] Cuckoo Sandbox. (2021). "Cuckoo Sandbox Documentation".

[7] “https://fpt-is.com/goc-nhin-so/lo-hong-trong-foxit-pdf-reader-cho-phep-thuc-thi-ma-tuy-y/”