VIET NAM NATIONAL UNIVERSITY HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

An Toàn Mạng Máy Tính

Báo cáo đồ án cuối kỳ

A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities

---

Giảng viên hướng dẫn: Nghi Hoàng Khoa

HO CHI MINH CITY, JANUARY 2023

Thành viên & Phân công

STT	Họ và tên	MSSV	Phần trăm đóng góp
1	Hồ Minh Trí	20522049	100%
2	Trần Hoài Rin	20521830	100%
3	Phan Võ Thiên Trường	20522091	100%

Contents

1 Overview

Đầu tiên, ta đến với việc giải thích từ viết tắt APT.

APT là tên viết tắt của Advanced Persistent Threat - thuật ngữ rộng dùng để mô tả một chiến dịch tấn công, thường do một nhóm các kẻ tấn công, sử dụng những kỹ thuật tấn công nâng cao để có thể hiện diện và tồn tại lâu dài trên mạng Internet nhằm khai thác dữ liệu có độ nhạy cảm cao.

Thông qua định nghĩa trên ta có thể tóm gọn APT gồm các mục tiêu sau:

- Theo đuổi các mục tiêu của mình lặp đi lặp lại trong một khoảng thời gian dài.
- Quyết tâm duy trì mức độ tương tác cần thiết, để thực hiện các mục tiêu của đã đề ra.
- Thích nghi với những nỗ lực của người bảo vệ để chống lại nó.

2 APT Attacking Methods

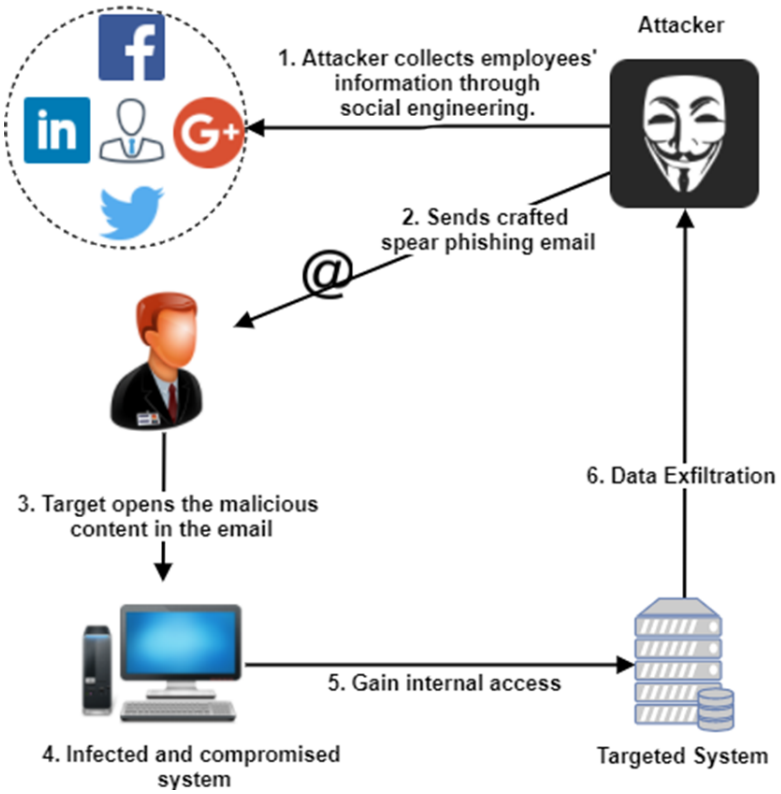

Figure 1: Tóm gọn lại 5 stage của APT attack

2.1 Giai đoạn 1: Reconnaissance

Nghiên cứu mục tiêu, thu thập thông tin về mục tiêu:

- Có thể bao gồm các thông tin về nhân viên như lối sống, thói quen, sở thích và các website họ thường vào nhưng không cần chi tiết.
- Chi tiết về cơ sở hạ tầng mạng, hệ thống như các loại switch, router, anti-virus tool, tường lửa,...
- Giúp attacker không chỉ tiến tới stage 2 establish foothold mà còn có thể thâm nhập sâu hơn vào network.

Việc này yêu cầu các kỹ năng: social engineering, do thám thông qua site, port, service scan.

2.2 Giai đoạn 2: Establish Foothold

- Tận dụng những lỗ hổng từng được công bố: CVE, OSVDB, NVD, lỗ hổng được chia sẻ trên các diễn đàn ở Deepweb, Darkweb.
- Malware: thông qua Spear-phising, USB, web download.
- Spear-Phishing: Email giả mạo có chứa thành phần độc hại.
- Zero-day: rât ít/hiếm khi xuất hiện những lỗi dùng zero-day này vì số lượng các lỗi này rất ít và các nhà phát hành cũng thường xuyên đưa ra các bản vá ngay lập tức khi phát hiện lỗi hoặc tạm thời ngừng hệ thống để đưa ra các biện pháp.
- Web Download:
  - Malicious websites.
  - Watering-Hole Attack: Thay vì giả dạng hay lừa đối tượng mở/truy cập vào mail/web khả nghi, phương pháp này chèn trực tiếp mã độc hại vào web mà đối tượng hay lui tới.

2.3 Giai đoạn 3: Lateral Movement/Stay Undetected

Thâm nhập sâu hơn vào hệ thống, mạng nội bộ tìm cách truy cập vào các dữ liệu, tài nguyên quan trọng. Trong stage này, hacker có thể:

- Ăn cắp dữ liệu chứng thực (đa số) bao gồm password, hash,... có thể được thu thập thông qua các tools:
  - Keylogger (password)
  - Mimikatz (thông dụng nhất)
  - Windows Credential Editor (WCE)
  - Riêng đối với thu thập thông tin chứng thực của Windows: phương pháp trích xuất và phân tích Windows Local Security Authority (LSA)
- Quăng malware và các tools khác vào hệ thống, các thiết bị.
- Chiếm các đặc quyền trong hệ thống.

2.4 Giai đoạn 4: Exfiltration/Impediment

Ở Stage này tuỳ vào mục đích của attacker ta có thể thấy ở một vài trường hợp:

- Ăn cắp dữ liệu (thông qua C&C).
- Phá hoại hệ thống.

2.5 Giai đoạn 5: Post Exfiltration/Post-Impediment

Và ở bước cuối cùng này, việc còn lại là tùy thuộc vào các nhà tài trợ của các attacker, họ sẽ quyết định là tiếp tục việc tấn công hay tiếp tục khai thác thông tin hoặc rút lui khỏi đó.

Trước khi rút lui, họ sẽ luôn thực hiện việc xóa dấu vết của mình ra khỏi hệ thống nhiều nhất có thể.

3 Case Study

Các cuộc tấn công APT đã xảy ra trước cả khi thuật ngữ APT ra đời.

APT Attack	**Date**	**Goal**	**Attack Vectors Used**
Titan Rain	2003 - 2005	Steal Organization Data	Social Engineering, Backdoors
Hydraq	2009 - 2011	Steal Organization Data	Social Engineering, Phishing, Backdoors, Zero-Day exploits
Stuxnet	2009 - 2012	Impede Critical Components	Malware via USB devices, Zero-Day Exploits, Backdoors
RSA SecureID Attack	2011 - 2011	Steal Organization Data	Spear-Phishing, Zero-Day Exploits, Backdoors
Carbanak	2013 - 2015	Steal Money	Social Engineering, Spear-Phishing, Backdoors, Key Loggers, Form Grabbers, Video Captures of Victim's Activities, Remote Administration Tools

Figure 2: Phân tích các trường hợp đã xảy ra

Như vào năm 2003, một loạt cuộc tấn công mạng đã diễn ra tại những nhà máy sản xuất vũ khí của Hoa Kỳ với mục tiêu đánh cắp các thông tin nhạy cảm. Chúng vẫn bị phát hiện còn tồn tại cho đến cuối năm 2015. Cuộc tấn công này là APT mặc dù lúc đó chưa có giới từ nào để diễn ta nó vào lúc đó và cuộc tấn công này có tên gọi là "TitanRain".

Tiếp đến vào năm 2009, sự hình thành dần dần của APT: chúng ta có cuộc tấn công APT lớn được xảy ra với tên gọi là "Operation Aurora - Hydrag". Cuộc tấn công sử dụng một số thành phần, phần mềm độc hại được mã hóa trong nhiều lớp để không bị phát hiện càng lâu càng tốt(Trojan) và phần mềm này được phát hiện sử dụng khai thác zeroday trong Internet Explorer(CVE-2010-0249 và MS10-002) và lỗ hổng (CVE-2009-1862) đối với 1 số công ty để có thể xâm nhập vào. Theo McAfee, mục tiêu chính của cuộc tấn công là giành quyền truy cập và thực hiện khả năng sửa đổi đối với kho lưu trữ mã nguồn tại các công ty công nghệ cao, an ninh và bảo mật, nhà thầu quốc phòng.

Cũng vào năm tiếp đó 2009, cuộc tấn công "Stuxnet" diễn ra với sự bắt đầu của 1 loại virus có sử dụng 4 lỗi zero-day để lây lan, che dấu sự tồn tại của chính nó, phá hoại, sao chép, với mục đích phá hoại chương trình hạt nhân của Iran, nhưng ngoài iran nó cũng đã tàn phá nhiều nơi trên thế giới so với mục đích ban đầu được tạo ra.

Năm tiếp theo nữa 2011, một số email đã được gửi cho nhà tuyển dụng với tệp tin excel được kèm theo, khi mở tệp này nó sẽ khai thác lổ hổng zero-day(CVE-2011-0609) của Adobe flash player để cài đặt một cửa hậu từ đó khai thác thông tin của công ty đó. Với việc này hững kẻ tấn công bắt đầu thu thập thông tin đăng nhập của một số nhân viên nhằm tiếp cận hệ thống mục tiêu nơi chúng thực hiện leo thang đặc quyền, đánh cắp dữ liệu và tệp, nén và mã hóa chúng trước khi gửi chúng đến trung tâm điều khiển và chỉ huy từ xa của chúng thông qua ftp.

Vào năm 2013, sự hoàn thiện của APT ngày nay, một dạng email lừa đảo giống như đã nói về cuộc tấn công năm 2011 nhưng tinh vi hơn, khi nhân viên mở các email đó, chúng sẽ khai thác các lỗ hổng Microsoft office(CVE-2012-0158, CVE-2013-3906 và CVE-2014-1761) khiến mã độc trong tệp đính kèm có khả năng cài đặt cửa hậu. Điểm mới trong cuộc tấn công này là các công cụ khác nhau mà chúng đã sử dụng và một giao thức nhị phân mà chúng đã thiết lập để liên lạc với các máy chủ chúng từ các máy nạn nhân. Người ta phát hiện ra rằng những kẻ tấn công đã nghiên cứu từng nạn nhân của chúng thông qua trinh sát nội bộ và sử dụng áp dụng phương pháp tấn công cụ thể cho nạn nhân đó. Họ đã tạo các giao dịch giả mạo trong cơ sở dữ liệu nội bộ của nạn nhân để che giấu các giao dịch chuyển tiền của họ. Carbanak dường như dừng lại vào năm 2015, nhưng sau đó người ta thấy rằng nó tiếp tục xuất hiện cho đến năm 2017 với các biến thể khác nhau.

4 APT Defense Methods

Nhóm tác giả đã đưa ra những phương pháp mà attacker dùng để vượt qua các lớp phòng thủ của hệ thống như:

- Dựa vào hệ điều hành và kiến trúc của hệ thống: Thông thường malware không thực thi trên hệ thống 64 bit vì những cách thức bảo mật được cập nhật trên đó làm việc khai thác trở nên phức tạp hơn.
- Các vector tấn công tiềm ẩn: Duqu 2.0 sẽ được nhắc đến ở dưới đây có phương pháp lây nhiễm chưa được xác định trước đó (sử dụng file Word, Excel bị nhiễm để xâm nhập).
- Thực thi lệnh và leo thang đặc quyền: thường là tận dụng các lỗi zero-day
- Thông qua truy cập mạng: Malware thường giao tiếp qua port 80, 443, 22 (TCP). Đầu ra của những port này thường cho phép qua mặt hệ thống kiểm duyệt truy cập mạng. Từ đây malware sẽ thành lập kênh giao tiếp C and C về hạ tầng của bên tấn công.
- Qua mặt IDS và đầu cuối của các chương trình diệt virus: Stuxnet, Duqu đều được thiết kế để phát hiện và né các chương trình quét virus bằng cách trang bị cho chúng một danh sách phương pháp né tránh kiểm duyệt. Hơn nữa, Duqu còn mã hóa đường truyền mạng của chính mình (C and C server) để né cả Network Intrusion Detection System (NIDS). Các hệ thống như NIDS phụ thuộc nhiều vào pattern matching (signature based detection)

- Mã hóa, che giấu tung tích: XOR encryption đóng vai trò quan trọng trong việc phát hiện và phân tích mã độc phức tạp (packing). Việc dùng kĩ thuật này làm cho file mã độc không thể bị phân loại nguy hiểm. Tuy nhiên việc này cũng là một dấu hiệu để nhận biết trong quá trình phân tích hành vi của mã độc.

Từ đây họ đã tổng kết được những phương pháp phòng chống được chia làm 3 phương diện dưới đây.

4.1 Monitoring Methods

Cách đơn giản nhất để phòng ngừa APT là giám sát mọi thứ trong hệ thống ở nhiều mức độ khác nhau để không tạo kẽ hở cho kẻ gian xâm nhập. Chia thành 3 mảng như sau.

4.1.1 Disk Monitoring

Mỗi thiết bị đầu cuối nằm trong hệ thống mạng của tổ chức phải luôn được kiểm soát trước các hoạt động bất thường của chương trình chống virus, tường lửa hay các bộ lọc. Cập nhật liên tục cho các phần mềm thuộc các thiết bị này cũng là một cách giảm rủi ro bị xâm nhập.

Ngoài ra việc kiểm soát tần suất sử dụng CPU cũng sẽ giúp phát hiện bất thường xảy ra trong hệ thống.

4.1.2 Memory Monitoring

Một trong những cách mà các phần mềm độc hại tránh mặt các chương trình phòng vệ chính là thực thi trên bộ nhớ của thiết bị thay vì từ một file execute (fileless malware). Ưu điểm của loại mã độc này là chúng không phát sinh ra bất kì tiến trình nào chạy nền nên việc truy vết nếu như chúng ta không quản lý tần suất sử dụng bộ nhớ hệ thống.

Năm 2015, Duqu 2.0 đã lây nhiễm vào Kaspersky Lab chứa Virus Stuxnet. Virus này nhắm vào PLCs(Programmable logic controller) bộ điều khiển logic khả trình (bộ điều khiển lập trình) là thiết bị điều khiển lập trình được (khả trình) cho phép thực hiện linh hoạt các thuật toán điều khiển logic thông qua một ngôn ngữ lập trình. PLC dùng để thay thế các mạch relay trong thực tế. Khi có sự thay đổi ở đầu vào thì đầu ra sẽ thay đổi theo.

Trong cuộc tấn công này, Stuxnet đã nhắm vào SCADA (hệ thống điều khiển giám sát và thu thập dữ liệu) và ngta cho rằng đây chính là nguyên nhân gây ra thiệt hại cho chương trình hạt nhân của Iran.

Korkin và Nesterow đã trình bày về cách phát hiện Zero day malware trong memory dump. Bài báo của họ đề cập tới việc sử dụng card đồ họa hiện đại hoặc kích hoạt CUDA trên GPU để phát hiện mã độc trên bộ nhớ. Bài báo nói về các driver ẩn có quyền hạn cao có khả năng tự che giấu chính mình và có thể làm ảnh hưởng đến hệ điều hành và chương trình diệt virus.

Các driver ẩn thường dùng cho spyware các cuộc tấn công này xảy ra thường xuyên làm cho việc kiểm soát các driver ẩn thành một nhiệm vụ cấp bách của an toàn thông tin.Theo Blunden trước khi một driver chế độ kernel khởi động, file exe của nó sẽ được đưa lên bộ nhớ và thông tin của driver được thêm vào nhiều danh sách liên kết của hệ điều hành và sau đó hàm DriverEntry sẽ được thực thi.

Lấy 3 driver A, B, C làm ví dụ. Hình trên chứa 3 driver đã được tải lên và nội dung của bộ nhớ ảo khi load chúng. Trên cùng là 2 danh sách liên kết với 3 cấu trúc driver tượng trưng cho 3 driver. Dưới cùng là 3 file exe của driver. Tất cả chúng sẽ được dùng như chữ kí để chúng ta phát hiện các driver.

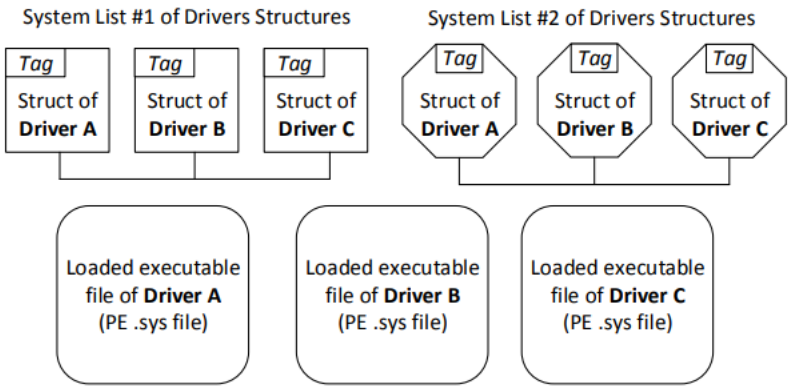

Figure 3: Quá trình khởi động của driver

Theo Russinovich, Solomon và Ionescu tool trên Window có một trong các danh sách như trên để nhận thông tin về driver gọi là NtQureySystemInformation cùng với Module SystemModuleInformation trong lớp information. Tuy nhiên kĩ thuật list này lại bị qua mặt bởi những kĩ thuật anti forensic. Cũng theo Blunden, bằng cách hủy liên kết cấu trúc của driver với danh sách kể trên thì driver này sẽ bị ẩn khỏi tool. Kĩ thuật này được gọi là Direct Kernel Object manipulation (DKOM) và nó dẫn tới 2 hậu quả: thứ nhất là tool build sẵn không còn phát hiện dc driver nữa. Thứ hai là Window với các driver đã bị ẩn đi vẫn hoạt động bình thường.

4.1.3 Packet Monitoring

Cách giám sát này dựa vào việc APT giao tiếp qua C and C nhiều lần nên việc giám sát packet đến và đi ở thiết bị đầu cuối có thể phát hiện được hoạt động bất thường trong hệ thống.

Villeneuve và Bennett cho rằng đôi khi những kĩ thuật khai thác cổ điển và mã độc đơn giản, attacker cũng có thể tấn công vào hệ thống thông qua đường truyền mạng. Vậy nên họ đi tới kết luận rằng giám sát đường truyền mạng có thể giúp phát hiện ra APT Attack. Liên hệ thực tế một chút, nhiều chiến dịch tấn công loại này được ghi lại là có thể bị phát hiện bằng cách phân tích đường truyền mạng mà attacker dùng để tạo C and C communication. Một số chiến dịch dài hơi nhưng ít tiếng tăm như Taidoor, IXESHE, Enfa (Lurid) cũng bị phát hiện thông qua tầng mạng:

- Taidoor là một dạng chương trình trojan cho phép tin tặc thực hiện đòn tấn công và kiểm soát máy tính của các nạn nhân từ xa (RAT) 2008.
- Công ty phần mềm an ninh Nhật Trend Micro Inc thông báo vừa phát hiện một chiến dịch tinh vi trên mạng nhằm ăn cắp thông tin cá nhân. Thủ phạm sử dụng phần mềm lừa đảo có tên IXESHE. Phần mềm độc này đã nhiễm vào máy tính của chính phủ các nước ở Đông Á cùng một số công ty máy tính, viễn thông của Đài Loan, Đức hoạt động tại châu Á.
- Enfal là loại malware được sử dụng trong đợt tấn công LURID, là đợt tấn công mà Trend Micro đã có một báo cáo phân tích cụ thể vào tháng 9 năm 2011.

Các chiến dịch trên đều tạo C and C bằng các giao thức đã biết như HTTP trên port 80, 443 (một cổng duyệt web được sử dụng để bảo mật thông tin liên lạc của trình duyệt web hoặc các dịch vụ HTTPS), 8080 (thường được sử dụng cho các máy chủ web). Attacker thường dùng những port này vì chúng được mở trên mức của tường lửa. Tuy nhiên, họ thường sử dụng những port này để bypass những luồng không hợp lệ như gửi gói khác HTTP trên port 80 và gói phi HTTPS trên port 443. Việc này có thể gây chú ý để điều tra sau này.

Giám sát thời điểm và kích thước gói tin cũng là 1 khía cạnh khác để phát hiện ra APT attack. Việc này là do khi thiết lập C and C, malwares thường gửi đi "beacon" - một gói tin giao tiếp với C and C server vào khoảng thời gian đã định.

Thêm vào đó, giám sát thời điểm sử dụng DNS Request hay URL cũng có thể phát hiện được APT vì các malware dù sử dụng HTTP thì chúng vẫn thường gửi request sử dụng API(Application Programing Interface - phương thức, giao thức kết nối với các thư viện và ứng dụng). Bằng cách phân tích HTTP header có thể phân loại API lạ với các hoạt động web bình thường.

Marchetti đã đưa ra một framework giúp phát hiện hàng nghìn host gồm các host có biểu hiện bất thường. Phương pháp của họ sẽ phân tích thông qua việc tập trung vào một số lượng host nhất định trong hàng ngàn host với mục đích loại bỏ host mà APT dùng để thực hiện C and C. Họ đưa ra một danh sách top các host bất thường mà APT sử dụng bằng cách giám sát khi APT vượt qua host nhiều lần rồi đem kết quả này so sánh với kết quả trong quá khứ. Cách này còn có hiệu quả với cả giao thức được mã hóa vì payload không được kiểm tra. Framework này liên quan tới việc thu thập và lưu trữ flow, extract feature, chuẩn hóa số liệu, tính toán điểm số rồi đánh giá.

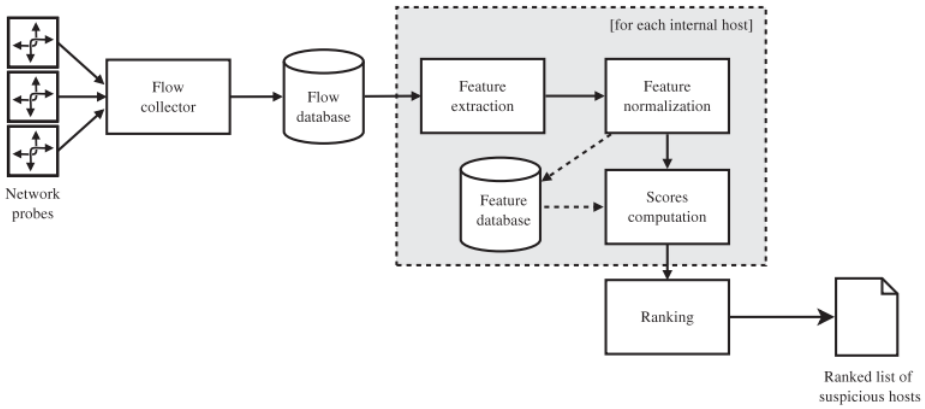

Figure 4: Mô hình framework của Marchetti

McCuskr cũng sử dụng biện pháp tương tự, qua việc theo dấu nhiều đối tượng mạng như host, hostgroup, và mạng để xem rằng chúng có bị đe dọa hay không. Hệ thống của ông ấy chia thành 5 lớp thực hiện 5 quá trình từ thu thập dòng dữ liệu cho tới việc tách thuộc tính. Sau khi tạo ra được không gian mẫu, ông dùng mô hình DL với 3 lớp cho cả mục đích học giám sát và bán giám sát để phát hiện biểu hiện lạ.

4.1.4 Code Monitoring

Việc này là hiển nhiên vì không một chương trình nào được tạo ra với số bug bằng không cả, nó chỉ nằm vùng chờ thời ở đâu đó trong chương trình thôi. Code sạch đã khó, việc khiến nó vẫn còn sạch trên những môi trường hoạt động khác nhau càng khó hơn chưa kể đến những bug mới chưa từng được phát hiện.

Những rủi ro từ mã nguồn như này thường được phát hiện thông qua các phương pháp phân tích tĩnh như Taint Analysis (là trường hợp đặc biệt của cách phân tích dòng dữ liệu, nó theo vết dữ liệu đã được đánh dấu xuyên suốt các con đường lan truyền dữ liệu của ứng dụng) hay Data Flow analysis. Thêm vào đó việc giám khác chương trình khi nó đang chạy để xem biểu hiện của nó cũng như chắc chắn rằng nó đang hoạt động đúng với ngữ cảnh đang được sử dụng kể cả khi nhận vào đầu vào không mong muốn, chắc chắn rằng nó sẽ không làm lộ hay hư hại vùng nhớ mà nó không có quyền truy cập đến.

Các phương pháp nêu trên giúp phát hiện tấn công sớm hơn trước khi nó lan ra nhiều hệ thống khác nhau.

4.1.5 Log Monitoring

Bản ghi không chỉ là một phần quan trọng trong pháp chứng mà còn có thể giúp phát hiện cũng như ngăn chặn chiến dịch APT từ rất sớm. Những bản ghi hữu ích như sử dụng bộ nhớ, sử dụng CPU, thực thi app, bản ghi hệ thống có thể cung cấp một số thông tin giúp ngăn chặn kẻ tấn công.

Bohara cùng nhóm tác giả của bài báo về kết hợp và phân cụm dữ liệu dám sát đã đưa ra một hướng tiếp cận trong việc phát hiện xâm nhập bằng cách kết hợp bản ghi của host và của mạng để tìm hành vi đáng ngờ. Từ dữ liệu đó, họ chọn ra 4 thuộc tính định danh, đường truyền, dịch vụ và phẩm quyền với mục đích làm giảm số liệu dư thừa để những thuộc tính k liên quan đến việc phân cụm sẽ được loại bỏ. Phương pháp này của họ có ap dụng học không giám sát để phát hiện bất thường mà không làm ảnh hưởng tới hệ thống.

Liên hệ với bên trên một chút, Shalaginov và các cộng sự phân tích bản ghi DNS để tìm ra các gói tin giao tiếp "beacon" giữa các host bị lây nhiễm và những tên miền đáng nghi. Về căn bản họ tin rằng việc tải mã độc vào máy để tạo một chỗ đứng trong hệ thống sẽ yêu cầu mở một kênh giáo tiếp C and C. Hành động này sẽ để lại một bản ghi chính nó trong dòng truyền mạng và bản ghi DNS. Nhóm tác giả đưa ra một phương pháp để phân tích bản ghi DNS nơi mà họ xác nhận rằng các host bị lây nhiễm sẽ giao tiếp C and C nhiều lần trong ngày.

Niu cùng các cộng sự cũng đã đưa ra phương pháp liên quan đến việc phân tích bản ghi DNS để phát hiện mã độc và hoạt động C and C chỉ là bản ghi này đến từ thiết bị di động. Họ trích xuất ra 15 thuộc tính được phân loại theo những thuộc tính phổ biến như: DNS request-answer, domain-based feature, time-based feature, whois-based feature.

Một thử thách cho việc giám sát bản ghi là việc ở đây có quá nhiều dữ liệu để xem và phân tích để phát hiện tấn công. Một nhóm tác giả đã đưa ra một hướng tiếp cận khác là phân tách thông tin từ những bản ghi bị bẩn. MÔ hình bao gồm 3 lớp: Beehive, một phương pháp sử dụng bản ghi DHCP server, bản ghi kết nối VPN từ xa, bản ghi ủy quyền và cả bản ghi quét virus để trích xuất các thuật tính dựa vào địa chỉ đích, host, các chính sách trên hệ thống và đường truyền sau đó phân cụm các thuộc tính trên theo thuật toán phân cụm K-means để xác định host nào bất thường.

Bhatt cùng các cộng sự cũng thảo luận về kill chain attack model và cũng đồng thời đưa ra giải pháp hoạt

động trên cấu trúc phân tầng. Với cấu trúc này, attacker có thể thực hiện ít nhất một lần tất cả các bước tấn công để vượt qua một lớp bất kì. Giải pháp phòng ngừa cách thức này có tỉ lệ thành công khá thấp khi phải tìm ra những rủi ro cơ bản nằm rải rác khắp các lớp, thứ mà việc sử dụng những lỗi đã được tìm ra trước đó ở lớp khác có thể không có tác dụng gì cho lớp này.

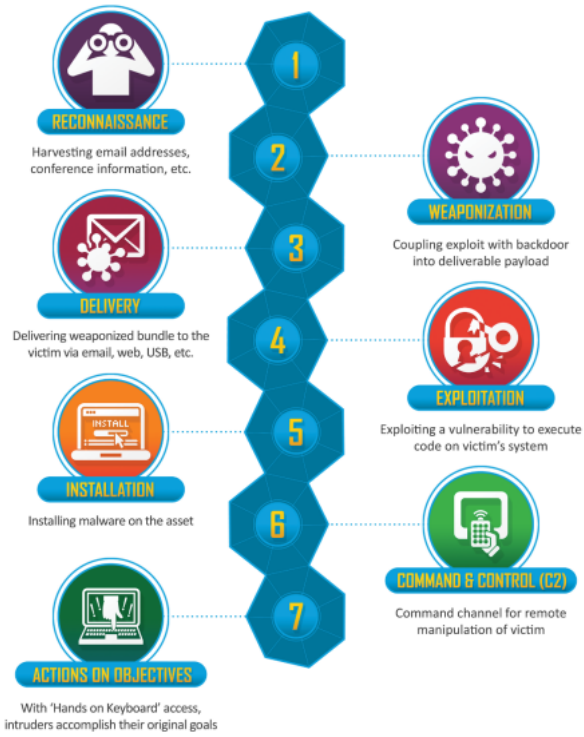

Figure 5: Kill chain attack model

Kill chain cũng là một loại tấn công nhiều công đoạn gồm 7 bước:

- Info gather: chọn mục tiêu, thu thập thông tin về mục tiêu.
- Weaponization: phát triển code khai thác để khai thác vào lỗ hổng đã tìm ra ở bước trên.
- Delivery: gửi code khả nghi vào mục tiêu.
- Exploitation: thực thi code thông qua lỗ hổng tìm được.
- Installation: cài đặt Remote Access Trojan(RAT) để duy trì quyền kiểm soát ở máy nạn nhân.
- Control and command (C2): C and C.
- Actions: bắt đầu việc phá hoại hệ thống.

Nhìn chung nhóm tác giả đã cung cấp rất nhiều thông tin về những cách mà thế giới đang sử dụng để giám sát hệ thống của mình khỏi những cuộc tấn công APT. Phần tiếp theo sẽ nói về một mảng khác của việc phòng chống APT đó là phát hiện xâm nhập.

4.2 Detection Methods

Cách thứ 2 là phát hiện ra những sự bất thường của hệ thống bằng những phương pháp có mục đích hoặc so sánh với những dữ liệu đã có sẵn từ đó phát hiện ra bất thường.

4.2.1 Anomaly Detection

Anomaly Detection là hát hiện ra những điều bất thường của hệ thống thông qua những phương pháp.

4.2.1.1 Approaches and Methods

Ở trong phần này, tác giả đã tiến hành đưa ra nhiều phương pháp để phát hiện ra sự bất thường, như là thông qua:

- Công cụ web
  - Cho phép ta lưu lại lịch sự những sự kiện đã xảy ra(IDS/IPS, cpu used, bộ nhớ,...).
  - Cho phép ta xem ở các dạng biểu đồ khác nhau(biểu đồ, sơ đồ cây, biểu đồ theo từng khu vực cụ thể, tổng quát tất cả).
  - Cho phép ta tìm kiếm dựa trên IP.
- Các phương pháp toán học
  Phân ra các loại bất thường đặc trưng:
  - Phát hiện bất thường dựa trên trình tự: cơ sở để phát hiện điểm bất thường dựa trên machine learning, trong đó các chuỗi dữ liệu huấn luyện và/hoặc kiểm tra được sử dụng để xác định điểm bất thường.
  - Phát hiện bất thường dựa trên trình tự con liền kề: có thể phát hiện hành vi của hệ thống cuối trong trường hợp tải xuống phần mềm độc hạihại.
  - Phát hiện bất thường dựa trên tần số mẫu: tần suất của các chuỗi cao hơn bình thường.

  Cũng qua phương pháp toán học này nói lên nhược điểm cách tiếp cận bất thường dựa trên chữ ký phổ(certificate key - signature), mang đến hai nhược điểm là:
  - Quy tắc xác định trước không đủ để phát hiện các cuộc tấn công.
  - Thiếu nhiều quy tắc xác minh trình tự hoạt động cụ thể.

  Có thêm kỹ thuật phát hiện dị thường theo lý thuyết thông tin và kỹ thuật phát hiện dị thường quang phổ hoạt động trong các cài đặt không giám sát.

- Phương pháp học máy
  - Dùng các Support Vector Machine(SVM) để phân loại dữ liệu đã được chuẩn hóa.
  - Dùng Artificial Neural Network(ANN) để chấp nhận các đầu vào và biến đổi đến khi đạt được đầu ra yêu cầu.
  - Dùng Fuzzy Logic(FL) để phát hiện các sự kiện.
  - Dùng Genetic Algorithm(GA) để tự tìm tòi, học hỏi.
  - Dùng các kỹ thuật phân loại, tìm kiếm các loại dữ liệu khác.
- Network(hệ thống, công cụ, phân tích)
  Như tựa đề đã nói trên cách này là thông pha qua các công cụ, hệ thống, hướng phân tích khác nhau liên quan đến sự bất thuờng của mạng, từ đó phát hiện ra sự bất thường.
- Bản chất của dữ liệu
  Thông qua bản chất của dữ liệu đầu vào và ra của loại bất thuờng, nhãn dữ liệu có sẵn. Từ đó chỉ ra bản chất của các thuộc tính dữ liệu đầu vào quyết định khả năng áp dụng các kỹ thuật phát hiện bất thường một cách tốt nhất.

4.2.1.2 Application to APT Detection

Khi nói về việc áp dụng phương thức Anomaly detection có thể thấy rằng nó giúp ích rất nhiều trong việc ngăn chặn các cuộc tấn công APT nhưng với việc các biến thể của malware được phát minh mới mỗi ngày. Còn công nghệ ta dùng để phát hiện và ngăn chặn không phải lúc nào cũng có thể bắt kịp với malware dẫn tới việc cần có những phân tích viên có kỹ năng trong việc phát hiện chúng và tìm ra phương hướng giải quyết.

Tuy nhiên, giữa 2 việc đó tồn tại một khoảng trống về thời gian đủ để attacker có thể thâm nhập vào network.

Để có thể phát hiện càng sớm càng tốt các cuộc APT attack. Việc sử dụng phương pháp học máy trong phân tích, phát hiện anomaly cũng như việc monitor, học, train và update model có thể rút ngắn khoảng trống đã được nêu cũng như nó có thể phát hiện những thay đổi nhỏ mà ít ai để ý.

Một số model học máy đã xuất hiện: Perceptrons, Neural Networks, Centroids, Binary Decision Tree, Deep Learning,...

*Khó khăn: Để phát hiện APT, chỉ một phương thức anomaly detection sẽ không đủ. VD, để có thể phát hiện việc sử dụng bộ nhớ bất thường của một process trong hệ thống. Hệ thống tìm kiếm cần phải biết log sử dụng memory của chính process đó ta cần model semi-supervised/supervised. Mà để tìm kiếm các anomaly behavior liên quan trong các process khác trong hệ thống ta cần model unsupervised.

Cái khó trong việc phát hiện anomaly trong việc ứng dụng học máy là false positives và false negatives, đặc biệt trong trường hợp của model semi-supervised và unsupervised vì các dữ liệu thu thập được không phải lúc nào cũng giống nhau dẫn tới việc xử lý dữ liệu giữa normal và abnormal data không phải lúc nào cũng sạch.

Tại hình 6 này ta sẽ so sánh một số phương thức anomaly-based APT attack defense cùng với cách thức học và phát hiện của chúng.

Reference	Learning Approach	Anomaly Detection Method	Source of Data	APT stages
[1]	Semi-Supervised	Machine Learning	Network/Host logs	Establish Foothold and Lateral Movement
[2]	Supervised	Machine Learning	Network Traffic	Establish Foothold
[3]	Semi-Supervised	Statistical	Host-based logs	Accomplishing Foothold
[4]	Semi-Supervised	Machine Learning	Network Traffic	Accomplishing Foothold, Lateral Movement, Exfiltration
[5]	Supervised	Machine Learning	Malware Detection	Accomplishing Foothold
[6]	Supervised	Machine Learning	Network Traffic	Lateral Movement, Exfiltration
[7]	Supervised	Machine Learning	Network Traffic	Internal Exfiltration
[8]	Supervised	Machine Learning	Emails	Reconnaissance, Establishing Foothold
[9]	Supervised, Unsupervised	Machine Learning	Host/Network events	Not specific to a stage, established behavior profiling
[10]	Unsupervised	Machine Learning	Active Directory domain service logs	Lateral Movement, Exfiltration
[11]	Supervised	Statistical	Network traffic, Access Information	Maintaining Access, Lateral Movement, Data Exfiltration

Figure 6: So sánh giữa anomaly-based APT attack defense

1 Kim et al., sử dụng rule-based anomaly detection để phát hiện APT attack cà phương pháp này có 2 stage. Stage 1 là nơi sử dụng ML và decision tree để tạo ra behavior rule dựa trên data thu thập được. Stage 2 là nơi phương thức abnormal behavior detection được dùng để tạo ra feature description sử dụng MapReduce dựa trên big data và behavior rule đã tạo ở Stage 2.

2 Zhao et al., hệ thống phát hiện APT malware infection. Bằng cách sử dụng DNS động để tìm kiếm C&C và có 2 giai đoạn. Phát hiện malicious C&C dựa trên phân tích các IP có liên quan xem có bất kỳ traffic nào bất thường hoặc đáng nghi không. Các tác giả dùng thuật toán J48 decision tree để phát hiện malicious DNS cùng với phương thức signature based detection và anomaly based detection.

3 Friedberg et al., tác giả đưa ra phương thức tiếp cận anomaly based detection truyền thống bằng cách học normal behavior của hệ thống và báo cáo lại bất kỳ hành động bất thường nào được phát hiện. Thay vì dùng phương thức black-list họ sử dụng log data tạo ra bởi nhiều hệ thống khác nhau và các thành phần trong ICT network. Từ đó tạo ra một model hệ thống dùng để phát hiện và phân biệt được các log khác nhau thông qua các lớp sự kiện chứa các mối liên quan giữa các sự kiện.

4 Cappers và Wijk, sử dụng ML để phát hiện APT trong network bằng cách phân tích nội dung trong network traffic và phân tách chúng thành thuộc tính và message. Tuy nhiên, phương thức này gặp khó với việc mở rộng ra khi thuộc tính thường có nhiều giá trị khác nhau. Dẫn tới 2 trường hợp:

   – Nhiều thuộc tính sẽ phá vỡ mối tương quan.
   – Ít thuộc tính tăng khả năng bị thiếu mối tương quan giữa các thuộc tính.

5 Yuan, sử dụng DL cho malware detection. Tác giả tin rằng các thuật toán ML thông thường không hiệu quả vì tỉ lệ xuất hiện false positive cao với lý do được nêu rằng malware và các phần mềm hiện nay đều phức tạp và đa dạng cùng với dataset khả dụng bị hạn chế hoặc lỗi thời dẫn tới các thuật toán ML thông

thường không thể dự đoán được feature trong giai đoạn học. Model của tác giả trong paper có kết quả khả quan hơn mặc dù chạy lâu hơn.

6 Siddiqui et al, tác giả cũng nêu rằng ML thông thường không hiệu quả vì tỉ lệ high positive cao nên thay vào đó sử dụng thuật toán phân loại malware phân mảnh để giảm thiểu false positives và false negatives. Họ dùng thuật toán ML K-Nearest Neighbor và dataset từ 2 nguồn khác nhau bao gồm APT traffic và non-malicious traffic. Họ test dataset bằng cách sử dụng supervised KNN và correlation based fractal dimension approach. Kết quả thu được khả quan với việc giảm tỷ lệ false positives và false negative dựa trên fractal dimension có khả năng trích xuất thông tin ẩn.

7 Cappers and Wijk, không ủng hộ phương pháp deep packet inspection (DPI) và anomaly detection vì chi phí của chúng quá đắt đỏ. Thay vào đó họ đưa ra phương hướng phân tích network traffic sử dụng visualization và ML cho phép admin hệ thống giám sát từng bộ phận trong network traffic. Dựa trên iterative refinement of classifier parameters và giám sát các alert message (payload inspection), họ dùng pixel visualization thể hiện cấu trúc của một network message và giảm tỷ lệ false positive. Phương pháp này tập trung vào monitoring traffic dẫn tới việc phải giám sát nhiều bộ phận nhỏ của một traffic cùng lúc. -> Khó phát hiện threat và malicious traffic hơn.

8 Dewan et al., phương pháp phân biệt spear phishing và non spear phishing email. Họ trích xuất feature từ các spear phishing email từ hơn 14 tổ chức quốc tế bằng cách sử dụng feature extraction từ LinkedIn và cũng như thực hiện việc nghiên cứu qua dataset từ Symatec scanning service. Các tác giả thu được 9 feature từ LinkedIn profile phishing email và họ tìm ra rằng classifier performance tệ khi dùng với social feature với lý do là thông tin thu thập được từ LinkedIn hạn chế.

10 Hsieh et al., sử dụng một framework phát hiện APT thông qua monitoring active directory log data. Framework tập trung vào input active directory logs và thu thập dữ liệu từ chúng. Nhìn chung, framework tìm kiếm sự thay đổi trong behavior của user thông qua phân tích log data của họ. Tuy nhiên, performance chỉ đưa về 66% độ chính xác -> anomaly detection dựa trên phân tích active directory log có phần nào hạn chế vì thông tin thu thập được từ log. Tác giả gợi ý có thể kết hợp active directory log với các log khác để tăng cường độ chính xác.

11 Marchetti et al., sử dụng một framework tên AUSPEX để trợ giúp các phân tích viên trong việc phát hiện và ưu tiên các signal liên quan tới APT. Framework bao gồm các kỹ thuật khác nhau dựa trên phân tích big data và security intelligent, thu thập và kết hợp thông tin từ nhiều nguồn khác nhau: thông tin nội bộ từ network probe bên trong tổ chức và bên ngoài từ web, mạng xã hội và black list. Sử dụng network flow log và thông tin thu thập được. Các tác giả tập trung vào 3 stage của APT: 2, 3 và 4, và ưu tiên các client nội bộ thể hiện hành vi khác thường.

Tại hình 7, tóm ý lại những vai trò mà ML đóng trong từng stage, cũng như là các thách thức chúng phải đối mặt tại từng stage. Các VD dưới đây có tương ứng với từng stage:

- Spear phishing: sử dụng supervised ML để học những feature từ những email spam trước. Mail thường chứa text, URL, SDT, hình,... -> train ra classifier để predict spear phishing email.

Stage	AI/ML Role	AI/ML Techniques	Challenges
Reconnaissance	Clustering	Unsupervised	High Volumes of Data to Process
Establishing foothold	Pattern Matching (classification)	Supervised	High False Positive Rate
Lateral movement	Grouping similar activities (Clustering), pattern matching (classification)	Unsupervised & Supervised	Dealing with numerous event data
Exfiltration	Pattern Matching (classification)	Supervised ML	High False Positive Rate
Cover up	Pattern recognition (neural network)	Supervised & Unsupervised ML	Huge volumes of low-quality evidence

Figure 7: Vai trò của Machine Learning trong từng stage

- Malicious DNS domains: liên tục thay đổi IP của URL. Có thể bị phát hiện thông qua kiểm tra DNS log xem URL có IP trước đó hay không. Từ đó có thể phát hiện ra bao nhiêu domain chia sẻ chung IP. Trong một cuộc APT attack, malware có thể ẩn nấp trong nhiều lớp mạng uỷ quyền.VD, attacker có thể thay đổi malicious URL mỗi vài phút mà không hề ngăn chặn user truy cập vào nó. Ta có thể sử dụng cả supervised và unsupervised ML để tăng khả năng phát hiện APT ngay trong stage 2 (tốt hơn phương pháp black list).
- User Profiling: Cần kiểm tra xem trong user log của từng user có quyền đăng nhập trái phép tới các pattern hay không. Sử dụng kỹ thuật cluster để admin system có thể phát hiện xem user nào có được đặc quyền trái phép. -> Unsupervised Machine Learning (clustering) trong stage 3
- Moving Data Monitoring: Kiểm tra thất thoát dữ liệu, luồng dữ liệu được truyền đi đâu, dung lượng,... -> Supervised Machine Learning trong stage 4
- Anomalous behavior: sử dụng Supervised ML để học các hành vi bình thường trong tổ chức và kiểm tra khi một hành vi được thi và không hề khớp với dữ liệu ML đã học thì sẽ cảnh cáo.

4.2.2 Pattern Matching

Pattern Matching là một phương pháp cũ nhưng vẫn còn chỗ hữu dụng. Bằng cách quan sát behavior của một process hoặc là ứng dụng, có thể phát hiện malicious behavior.

12 Yan et al., sử dụng hệ thống phát hiện xâm nhập để phát hiện APT. Phương pháp dựa vào thông tin từ thông tin kiến trúc tầng cao thu thập được trong network traffic.

13 Giura và Wang đưa ra một model phát hiện APT với phương pháp áp dụng vào generic network của tổ chức. Phương pháp do tác giả đề xuất bao gồm 3 sự kiện.

- Candidate: tất cả sự kiện đều được tổ chức lưu trữ lại.
- Suspicious: sự kiện được báo cáo bởi cơ chế bảo vệ, hoặc là có mối liên hệ với các hành vi bất thường.
- Attack: thường được các hệ thống bảo vệ truyền thống nhằm vào phát hiện bất cứ hành vi tấn công có thể xác định.

Các sự kiện có mối liên kết với nhau thông qua nội dung của từng cái và rule (correlation rule, detection rule) để tìm ra các mối đe doạ tiềm tàng. Từ đó risk level và confidence indicator được dùng để đánh giá mức độ nguy hiểm của threat.

4.3 Mitigation Methods

Bên cạnh việc giám sát hệ thống để phát hiện xâm nhập sớm, xây dựng cơ chế phát hiện xâm nhập để có thể biết hệ thống đang bị đe dọa bởi những bug, phương thức tấn công nào thì việc khắc phục và làm giảm thiểu thiệt hại khi hệ thống đã bị xâm nhập cũng cực kì quan trọng.

Việc giảm thiểu thiệt hại và khắc phục lỗi cũng được chia làm hai phương thức chính.

4.3.1 Reactive Methods

Phương pháp này xác định những ngữ cảnh tấn công khả dĩ dựa vào những lỗ hổng hiện tại của hệ thống và đưa ra những phân tích hướng tấn công mà attacker có thể dùng để xâm nhập hệ thống.

Graph analysis (Phân tích đồ thị) là một linh vực cực kì hữu ích cho việc phân tích những mạng phức tạp và phát hiện những cuộc tấn công tinh vi. Attack graph được dùng như là một tool để phát hiện multi-hop attack trong một mạng. Một attack graph có thể được biểu diễn như 1 tập G = N, E.

- Các node có thể được biểu diễn là N = Nf U Nc U Nd U Nr. Nf là các node thật. Ví dụ, access list hacl (VM1, 80, VM2, 5000) có nghĩa là VM1 và VM2 có thể giao tiếp thông qua 2 cổng 80 và 5000. Nc là cổng đã bị khai thác, ví dụ execCode(VM1, apache, user), có nghĩa là trên apache web server attacker có thể thực hiện code bằng leo thang đặc quyền. Nd chính là quyền hạn hiện tại (root, VM1) và Nr chính là node đích (root, DatabaseServer) có nghĩa là đã lấy dc quyền root tại server của database.

- Các cạnh được biểu diễn bằng tập những cạnh với điều kiện tiên quyết là E = Epre U Epost. Epre nằm trong tập (Nf U Nc) x (Nd U Nr) nghĩa là NC và Nf phải giao nhau để thu được Nd. Epost nằm trong tập (Nd U Nr) x (Nf U Nc) có nghĩa là điều kiện của Nd phải thỏa cả điều kiện của Nf và Nc.

Một attack graph có thể được sử dụng để nghiên cứu lối tấn công trong ngữ cảnh của APT bởi vì trình tự của sự kiện dẫn đến xâm nhập hệ thống diễn ra tuần tự. Một ưu điểm nữa của Attack Graph là sự dễ dàng trong việc ước tính hao tổn của việc tấn công và giá trị mà nó mang lại cho phía bên kia từ đó cũng tìm ra được cách đối phó cho phương pháp họ chọn để tấn công.

Bảo mật bằng cách phân tích attack graph có thể giúp phát hiện những khu vực nguy hiểm bậc nhất trong hệ thống và mức độ nghiêm trọng của các cuộc tấn công trước đây cũng đóng góp cho các kịch bản phòng tấn công APT. Dựa vào kiểu tấn công, mục đích tấn công và payload dùng tấn công phương pháp này có thể được áp dụng vào việc đánh giá bảo mật.

4.3.2 Proactive Methods

Kĩ thuật dựa vào những phương pháp có thể bịp attacker hoặc đổi môi trường bị tấn công khiến cho nó khó khăn hơn cho việc khai thác. Có hai kĩ thuật chính.

- Honeybot và Honeynet:
  Một trong những yếu tố khiến cho APT attack khác biệt là lượng nhân sự tham gia tấn công, nhân sự quá hùng hậu khiến cho loại mã độc và hình thức tấn công trở nên quá đa dạng và phức tạp khiến cho bên phòng thủ khá ngợp. Và thường thì họ sẽ dùng proactive method như một cú lừa để cho họ thời gian theo kịp những thứ kì lạ và mới mẻ.

  Trong phương pháp này, bên phòng thủ lừa bên kia bằng một loại tài liệu nhử hoặc tạo ra một hệ thống hay mạng mồi giống y hệt máy chủ thật nhưng nằm ngoài hệ thống chính để attacker tấn công vào đó. Bằng cách giám sát hoạt động của attacker trên hệ thống/mạng này có thể giúp bên phòng chống biết được thủ pháp mà attacker dùng để thông qua các lớp phòng ngự của hệ thống tổ chức.

  Bài báo của Bowen và các cộng sự đặt ra vấn đề rằng chính nội bộ tổ chức cũng chính là một nguồn gây ra APT attack. Nhóm tác giả đưa ra một phương pháp gài bẫy attacker muốn trích xuất dữ liệu để lấy thông tin nhạy cảm bằng cách đưa vào đó một cơ số dữ liệu giả khiến cho họ gặp khó khăn trong việc phân biệt đâu là thông tin quan trọng đâu là thông tin rác làm cho hành vi gây hại này cần thêm nhiều thời gian và công sức hơn. Đống dữ liệu giả đó được tạo và thêm tự động vào hệ thống nhử mồi để dẫn dụ attacker bằng những thông tin xác thực không có thật - thứ sẽ gây ra cảnh báo khi được dùng để khai thác sâu vào hệ thống. Cụ thể là họ sẽ đánh dấu vào mã binary của file đó để khi mà nó được thực thi, hệ thống sẽ đưa ra cảnh báo về hành vi bất thường.

  Tác giả phân loại mức độ tinh vi của attacker làm 3 mức thấp, trung, cao và sau đó đưa ra những cách mà attacker ở mức độ này hay sử dụng. Họ cũng giải thích cách mà tập tin giả được tạo ra ví dụ như embedded honey token, tài khoản đăng nhập, đầu ra của mạng giúp cảnh báo khi tập tin mồi được gửi đi hay giám sát ở host có thể phát hiện khi attacker thao tác vào file mồi, beacon được cài vào sẽ thông báo khi có kết nối remote server.

  Anagnoakis cùng cộng sự cũng đưa ra một framework giả dạng tận dụng máy ảo và mạng được dựng bằng phần mềm để tạo ra một môi trường giả mạo khó đoán và tự thích ứng cực tốt. Tuy nhiên, do sự hạn chế của công nghệ hiện tại không thể dùng SDN hay cloud để dựng một môi trường có độ chân thực cao làm cho phương pháp xây dựng mạng giả này không phát huy được tối đa lợi ích của mình.

  Nhóm tác giả trên cũng đưa ra một cấu trúc lai giữa mô hình honeypot xịn nhất và hệ thống phát hiện bất thường. Hệ thống này gồm nhiều monitor giám sát đường truyền mạng đến hệ thống đích. Đường truyền dị thường sẽ được chuyển tới một honeypot bóng để đánh giá độ gây hại của nó. Honeypot thế mạng đó chính là ví dụ tiêu biểu cho chương trình bảo vệ.

  Họ cũng kết luận rằng mô hình của họ có hiệu quả cao hơn so với hai mô hình khi dùng riêng lẻ bởi vì:

  - Nó làm giảm hiện tượng dương tính giả ở honeypot trong việc xác định xâm nhập.
  - Chỉ cần mồi con giống y bản gốc thì hệ thống sẽ được an toàn khỏi biện pháp tấn công đã bị đánh lừa trong một khoảng thời gian.
  - Bảo vệ chương trình khỏi bị tấn công từ phía khách hàng.
  - Dễ dàng thêm mới các cơ chế bảo mật.

- Moving Target Defense:
  Khác với Honeypot, phương pháp này được Crouse và những cộng sự cho rằng điều tiên quyết nằm ở việc đánh lừa là chuyển dời. Họ chỉ ra rằng phương pháp này hoạt động bằng cách chuyển bề mặt tấn công liên tục từ đó attacker sẽ không ở trong môi trường tĩnh và trạng thái dài hạn của mạng không được duy trì sẽ làm ảnh hưởng đến giai đoạn thăm dò của cuộc tấn công.

  Ví dụ sống cho MTD là network shuffling - map lại địa chỉ với mục đích làm cho kết quả quét ip trở nên vô dụng. Nhóm tác giả này còn đưa ra luận điểm rằng honeypot có thể dùng để lừa những attacker đang trinh sát một cách hiệu quả.

  MTD được chia làm 3 trường phái:

  - Shuffle: cho phép hệ thống mạng có thể tái sắp xếp nhiều lớp trong protocol stack (VM migration, topology Rearrangement, port hopping).
  - Diversity: phương pháp tạo ra một biến thể tương ứng với hệ điều hành hiện tại.
  - Redundancy: kĩ thuật cung cấp bản sao của ứng dụng hay tài nguyên mạng như là máy ảo giả mạo, proxies, đường truyền mạng.

  Còn một phương diện nữa để phân biệt các mảng này là dựa vào protocol stack:

  - Network level: thay đổi network topology (IP hopping, che dấu đường truyền).
  - Host level: thay đổi tài nguyên trên host, OS, đổi tên cấu hình...
  - Application level: thay đổi ứng dụng, source code, memory map, phiên bản phần mềm...

5 Evaluation and Challenges

Các phương pháp phát hiện thời nay đều dựa vào ML với 3 thành phần quan trọng sau: thu thập dữ liệu, trích xuất thuộc tính và testing. Tuy nhiên khó khăn ở đây là những dữ liệu thu được thông qua các ngữ cảnh được thực hiện trong môi trường được kiểm soát nơi mà không có sự gây nhiễu từ tác nhân ngoại cảnh khiến cho độ phủ dữ liệu không đủ rộng và sẽ để sót những lỗ hổng để attacker khai thác hoặc là lượng dữ liệu từ thực tiễn vẫn còn quá ít.

Kế đến là vấn đề chọn thuộc tính - việc mà sẽ gây tác động đến kết quả cũng như mô hình được huấn luyện thành. Thường thì dữ liệu sau khi được thu thập vẫn không thể nào dùng để huấn luyện ngay mà phải tốn thời gian để xử lí và trích lọc. Tuy rất khó khăn trong việc chọn thuộc tính nhưng tác giả cũng đã tổng hợp cho chúng ta một bảng thuộc tính hay được chọn nhất.

Bản thân APT Attack đã là một thử thách cho các kĩ sư an toàn thông tin trong việc phòng chống nó bởi vì có quá nhiều thứ tham gia vào chiến dịch này. Trong đó có những thành phần đặc biệt đáng chú ý như:

- Attacker kiên trì, giàu kĩ năng.
- Các cuộc tấn công kéo dài.

Mining Techniques	Common Features	Targeted APT Stage
Emails	terms, structure statistics, values of emal header fields like "From", "To", and "CC", email has an attachment or not, the domain name of an email address, the email sender's information, such as the writing style and the user name of the email sender [99]	Reconnaissance
Malware	strings, byte sequences, opcodes APIs/System calls, memory accesses, file system accesses, Windows registry, CPU registers, function length, PE file characteristics, and raised exceptions, network, AV/Sandbox submissions, and code stylometry [100]	Foothold (watering hole, spear phishing)
DNS logs	IP addresses, distinct domain names, number of queries at each domain name by time, authoritative answer, type of DNS packet requested, resource record time to live (i.e., high TTL values are likely indicators of malicious domains)	C&C communication
System logs	failure login attempts, source/destination IP addresses, service type, protocol type, CPU utilization, file system usage, health status, network flows, internal and external flows, running process	Lateral movement, C&C communication
Outgoing Network Traffic	source/destination port addresses, type of physical media, source/destination IP addresses, service type, protocol type, flow direction, bytes sent, average packet size, average received size, traffic flow ratio, interval of packets sent	Data exfiltration, C&C communication

Figure 8: Những thuộc tính thường được chọn

- Nhân sự công ty thiếu hiểu biết khiến cho mã độc bị tuồn vào hệ thống một cách vô ý.
- Trình độ công nghệ của mục tiêu bị khai thác lạc hậu.

References

1 H. Kim, J. Kim, I. Kim, and T.-m. Chung, "Behavior-based anomaly detection on big data," 2015.

2 G. Zhao, K. Xu, L. Xu, and B. Wu, "Detecting apt malware infections based on malicious dns and traffic analysis," IEEE Access, vol. 3, pp. 1132–1142, 2015.

3 I. Friedberg, F. Skopik, G. Settanni, and R. Fiedler, "Combating advanced persistent threats: From network event correlation to incident detection," Computers & Security, vol. 48, pp. 35–57, 2015.

4 B. C. Cappers and J. J. van Wijk, "Understanding the context of network traffic alerts," in Visualization for Cyber Security (VizSec), 2016 IEEE Symposium on. IEEE, 2016, pp. 1–8.

5 X. Yuan, "Phd forum: Deep learning-based real-time malware detection with multi-stage analysis," in Smart Computing (SMARTCOMP), 2017 IEEE International Conference on. IEEE, 2017, pp. 1–2.

6 S. Siddiqui, M. S. Khan, K. Ferens, and W. Kinsner, "Detecting advanced persistent threats using fractal dimension based machine learning classification," in Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics. ACM, 2016, pp. 64–69.

7 B. C. Cappers and J. J. van Wijk, "Snaps: Semantic network traffic analysis through projection and selection," in Visualization for Cyber Security (VizSec), 2015 IEEE Symposium on. IEEE, 2015, pp. 1–8.

8 P. Dewan, A. Kashyap, and P. Kumaraguru, "Analyzing social and stylometric features to identify spear phishing emails," in Electronic Crime Research (eCrime), 2014 APWG Symposium on. IEEE, 2014, pp. 1–13.

9 O. McCusker, S. Brunza, and D. Dasgupta, “Deriving behavior primitives from aggregate network features using support vector machines,” in Cyber Conflict (CyCon), 2013 5th International Conference on. IEEE, 2013, pp. 1–18.

10 C.-H. Hsieh, C.-M. Lai, C.-H. Mao, T.-C. Kao, and K.-C. Lee, “Ad2: Anomaly detection on active directory log data for insider threat monitoring,” in Security Technology (ICCST), 2015 International Carnahan Conference on. IEEE, 2015, pp. 287–292.

11 M. Marchetti, F. Pierazzi, A. Guido, and M. Colajanni, “Countering advanced persistent threats through security intelligence and big data analytics,” in Cyber Conflict (CyCon), 2016 8th International Conference on. IEEE, 2016, pp. 243–261.

12 X. Yan and J. Zhang, “Early detection of cyber security threats using structured behavior modeling,” ACM Transactions on Information and System Security, vol. 5, 2013.

13 P. Giura and W. Wang, “A context-based detection framework for advanced persistent threats,” in Cyber Security (CyberSecurity), 2012 International Conference on. IEEE, 2012, pp. 69–74.

14 Alshamrani, Adel, et al. "A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities." IEEE Communications Surveys & Tutorials 21.2 (2019): 1851-1877.