Testy Penetracyjne

Zbieranie informacji — Metody aktywne

Arkadiusz Chrobot

Katedra Systemów Informatycznych

20 marca 2024

Plan

Wstęp

Metody pasywne i aktywne mogą być stosowane naprzemiennie w procesie zbierania informacji. Zazwyczaj rozpoczyna go użycie metod pasywnych, a potem następuje przejście do aktywnych, które bezpośrednio wchodzą w interakcję z testowanym systemem. W ich trakcie jest sporządzana, a następnie zawężana lista urządzeń sieciowych, sieci i innych potencjalnych celów. Filtrowanie tej listy może odbywać się na podstawie różnych kryteriów, np. usług, które są uruchomione na poszczególnych urządzeniach. Na tym etapie testowania również wykorzystywane są specjalistyczne narzędzia.

Źródła informacji

Odkrywanie urządzeń sieciowych (ang. *host enumeration*) niekoniecznie musi ograniczać się do użycia oprogramowania skanującego sieć. Należy pamiętać, że nie jest ono niezawodne i może dawać zarówno wyniki fałszywie dodatnie, jak i fałszywie ujemne. Trzeba więc porównać jego wynik z innymi źródłami informacji, jakimi mogą być:

- centralne systemy zarządzające, np. Microsoft's Endpoint Configuration Manager,
- pliki konfiguracyjne, np. serwerów DHCP, tablice ARP, logi ruterów, wyniki polecenia `arp-scan`, itp.

Identyfikacja usług

Ustalenie jakie usługi (i jakie ich wersje) są uruchomione na urządzeniach połączonych w sieć jest istotną czynnością w testowaniu penetracyjnym. Dzięki tym informacjom można zidentyfikować podatności, które w tych usługach występują i wykorzystać je w fazie eksploatacji. Dodatkowo można w ten sposób zidentyfikować (ang. *fingerprint*) systemy operacyjne, pod których kontrolą te urządzenia pracują. One również mogą mieć luki zabezpieczeń przydatne z punktu widzenia pentestera. Przykładowo, jeśli na urządzeniu uruchomiona jest usługa SSH (zwykle port 22), to jego systemem operacyjnym jest zapewne kompatybilny z Uniksem (Linux, lub wariant BSD, np. NetBSD). Jeśli urządzenie posiada otwarte porty 139 (NetBIOS), 445 (Active Directory i SMB) oraz 3389 (RDP), to tym systemem prawdopodobnie będzie MS Windows.

Identyfikacja systemu operacyjnego

Otwarte porty i dostępne przez nie usługi nie są jednak wiarygodnym i dokładnym sposobem na ustalenie rodzaju i wersji systemu operacyjnego. Narzędzia, które potrafią przeprowadzić taką operację stosują dokładniejsze sposoby, które bazują na badaniu sposobu odpowiedzi na pakiety TCP i UDP (jakie opcje TCP system obsługuje, w jakiej kolejności wysyła pakiety i inne).

Identyfikacja usług

Do aktywnego odkrywania usług służą *skanery protów*. Porty 1–1023 są używane przez uprzywilejowane procesy w systemach uniksowych i nazywane *portami systemowymi* lub *dobrze znanymi* (ang. *well-known ports*). Z kolei porty 1024–49151 są portami *zarejestrowanymi*, co znaczy, że zostały przypisane określonym usługom przez IANA. Port 0 ma specjalne znaczenie i nie jest używany przez zwykłe usługi. Ogólnie dla usług są dostępne zatem porty 1–65535. Należy zaznaczyć, że niektóre, lub czasem nawet wszystkie, mogą być uruchomione na innych portach, niż te, które zwyczajowo się im przypisuje. Z tego względu skanery portów próbują utworzyć połączenie i uzyskać *baner* usługi, czyli ciąg bajtów (np. tekstu), który pozwoli poznać ich prawdziwą tożsamość.

Nmap

Nmap jest popularnym narzędziem do testów penetracyjnych, które często jest wykorzystywane jako skaner portów. Jest ono uruchamiane z wiersza poleceń. Najczęściej stosowanymi argumentami dla tego polecenia, oprócz adresu lub adresów IP, są:

- `-O` zidentyfikuj system operacyjny — wymaga uruchomienia z uprawnieniami użytkownika uprzywilejowanego;
- `-A` zidentyfikuj system operacyjny i uruchomione usługi;
- `-T` szybkość skanowania, wyrażona numerycznie lub słownie: 0 (paranoid), 1 (sneaky), 2 (polite), 3 (normal), 4 (aggressive), 5 (insane) — domyślnie stosowana jest wartość 3, większe pozwalają szybciej skanować, ale to skanowanie może być łatwiej wykryte przez mechanizmy zabezpieczające;
- `-p` numer portu lub portów, które mają być skanowane, można podać także nazwę usługi lub zakres protów, zamiast 1–65535 można użyć "*";

Nmap

`-oN` nazwa i format pliku wyjściowego, za tym argumentem należy podać nazwę pliku, użytym formatem będzie zwykły tekst, istnieją inne podobne argumenty;

-PN można także użyć `-Pn`, skanowanie bez użycia pakietów ICMP (ping);

-sA skanowanie z użyciem pakietów TCP ACK, ułatwia testowanie reguł zapór sieciowych i ustalenie, czy zapora zachowuje stan połączenia (ang. *stateful*);

-sT skanuje w poszukiwaniu usług korzystających z protokołu TCP;

-sU skanuje w poszukiwaniu usług korzystających z protokołu UDP;

-sS skanuje używając pakietów TCP SYN, czyli nie kończąc potwierdzania połączenia (ang. *handshake*), jest to szybkie i zazwyczaj trudne do wykrycia skanowanie, choć np. zapory sieciowe mogą sobie z tym radzić.

Nmap

Adres IP jest przekazywany do narzędzia `nmap` jako pierwszy. Może to być także zakres adresów `IP`, a nawet adres całej sieci w notacji `adres IP/CIDR`. Działanie tego narzędzia można przetestować skanując adres `scanme.nmap.org`, ale trzeba przestrzegać zasad podanych na tej stronie.
Slajd 11 zawiera wynik skanowania narzędziem nmap komputera RasberyPi 2 z systemem operacyjnym Raspbian, włączoną zaporą sieciową i uruchomionymi kilkoma usługami. Nmap został uruchomiony poleceniem:

```
nmap -A -p"*" -oN pi-firewall adres_IP
```

Narzędzie to domyślnie skanuje 1000 najczęściej stosowanych portów. Można to zmienić stosując argument `-p`. Ponieważ celem tego badania było skanowanie wszystkich portów, to wraz z tym argumentem została podana opcja `"*"`. Cudzysłów jest konieczny, bo bez niego znak `*` zostałby zinterpretowany przez powłokę, nie przez nmap.

Nmap

```
# Nmap 7.94SVN scan initiated Sat Mar 16 13:46:42 2024 as:
↪  nmap -A -p* -oN pi-firewall 192.168.XXX.YYY
# Nmap done at Sat Mar 16 13:46:45 2024 -- 1 IP address (0
↪  hosts up) scanned in 3.20 seconds
```

Użyte w eksperymencie Raspberry Pi 2 odpowiada na pakiety ICMP wysyłane przez polecenie `ping`, ale zapora sieciowa sprawdza również, czy nie są one używane do skanowania urządzenia i blokuje je, jeśli tak jest. Dlatego w tym przypadku nmap nie zwrócił praktycznie żadnych istotnych informacji. Uznał, że urządzenie jest wyłączone. W takim wypadku lepiej jest wyłączyć użycie pakietów ICMP argumentem `-Pn` (lub `-PN`). Rezultat skanowania z użyciem wspomnianej opcji przedstawiony jest na slajdzie 12.

Nmap

```
# Nmap 7.94SVN scan initiated Sat Mar 16 13:47:12 2024 as: nmap -PN -A -p*
↪  -oN pi-firewall-noping 192.168.XXX.YYY
Nmap scan report for raspberrypi (192.168.XXX.YYY)
Host is up (0.0023s latency).
Not shown: 8366 filtered tcp ports (no-response)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.0p1 Debian 4+deb7u7 (protocol 2.0)
| ssh-hostkey:
|   1024 c4:21:c8:78:35:52:ed:fc:66:14:ff:43:40:1f:48:60 (DSA)
|   2048 e9:98:23:1b:74:bd:fb:04:9c:30:12:af:b2:9f:01:e3 (RSA)
|_  256 b2:0d:10:e7:44:73:9d:c6:ea:d6:f0:df:b8:c2:2e:bf (ECDSA)
8080/tcp open  http    Jetty 9.4.22.v20191022
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: Jetty(9.4.22.v20191022)
|_http-title: Site doesn't have a title (text/html;charset=utf-8).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
↪  https://nmap.org/submit/ .
# Nmap done at Sat Mar 16 14:31:09 2024 -- 1 IP address (1 host up) scanned
↪  in 2636.68 seconds
```

Namp

Tym razem narzędziu udało się wykryć rodzaj sytemu operacyjnego (Linux) oraz dwie usługi, które są na nim uruchomione (SSH — port 22 — i serwer Jetty — port 8080) i kilka szczegółów z nimi związanych. Slajdy 14 i 15 zawierają wynik narzędzia Nmap, uruchomionego poleceniem[1] ze slajdu 8, ale dla komputera Raspbery Pi 2, w którym wyłączono zaporę sieciową.

[1] Jest jedna różnica — zakres portów został określony w inny sposób, ale w obu przypadkach jest taki sam.

Nmap

Nmap

```
# Nmap 7.94SVN scan initiated Sat Mar 16 14:48:56 2024 as: nmap -A -p 1-65535 -oN
↪  pi-nofirewall-noping 192.168.XXX.YYY
Nmap scan report for raspberrypi (192.168.XXX.YYY)
Host is up (0.00082s latency).
Not shown: 65528 closed tcp ports (conn-refused)
PORT      STATE SERVICE    VERSION
22/tcp    open  ssh        OpenSSH 6.0p1 Debian 4+deb7u7 (protocol 2.0)
| ssh-hostkey:
|   1024 c4:21:c8:78:35:52:ed:fc:66:14:ff:43:40:1f:48:60 (DSA)
|   2048 e9:98:23:1b:74:bd:fb:04:9c:30:12:af:b2:9f:01:e3 (RSA)
|_  256 b2:0d:10:e7:44:73:9d:c6:ea:d6:f0:df:b8:c2:2e:bf (ECDSA)
5432/tcp  open  postgresql PostgreSQL DB 9.1.20 - 9.1.24
| ssl-cert: Subject: commonName=raspberrypi
| Not valid before: 2015-10-16T17:03:06
|_Not valid after:  2025-10-13T17:03:06
|_ssl-date: TLS randomness does not represent time
5900/tcp  open  vnc        VNC (protocol 3.8)
| vnc-info:
|   Protocol version: 3.8
|   Security types:
|     VNC Authentication (2)
|     Tight (16)
|   Tight auth subtypes:
|_    STDV VNCAUTH_ (2)
6000/tcp  open  X11        (access denied)
8080/tcp  open  http       Jetty 9.4.22.v20191022
|_http-server-header: Jetty(9.4.22.v20191022)
|_http-title: Site doesn't have a title (text/html;charset=utf-8).
| http-robots.txt: 1 disallowed entry
|_/
```

Nmap

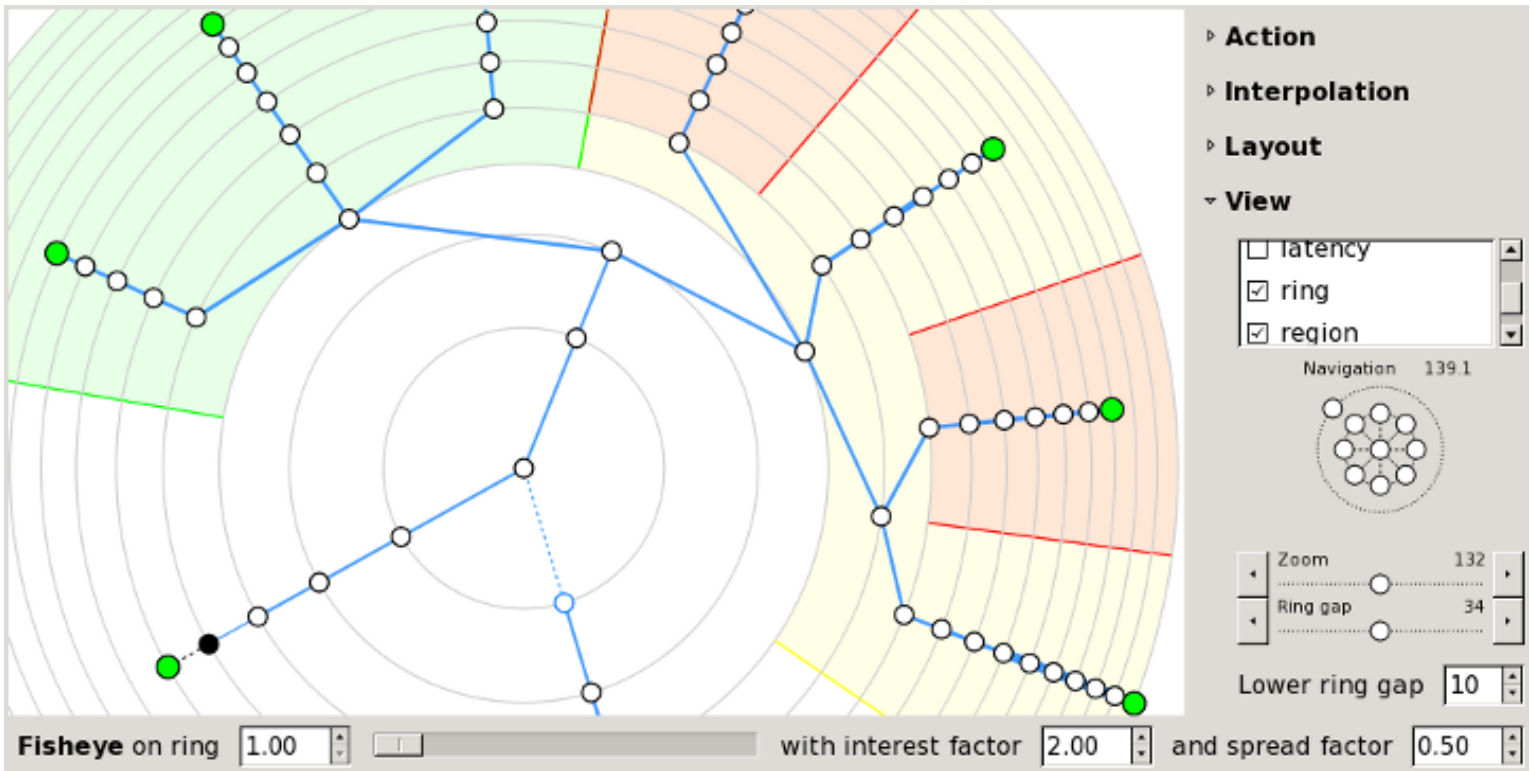
Nmap

```
9418/tcp  open  git?
45323/tcp open  http       Jenkins httpd 2.210
|_http-server-header: 192.168.XXX.YYY
|_http-title: Site doesn't have a title (text/plain;charset=UTF-8).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
Service detection performed. Please report any incorrect results at
↪  https://nmap.org/submit/ .
# Nmap done at Sat Mar 16 14:50:28 2024 -- 1 IP address (1 host up) scanned
↪  in 92.08 seconds
```

Jak łatwo zauważyć, tym razem Nmap odkrył dodatkowe usługi (PosgreSQL, X11, vnc, Jenkins, git) i podał związane z nimi szczegóły. Ważnym spostrzeżeniem jest, że szybkość skanowania wpływa na uzyskiwane przy pomocy programu Nmap wyniki. Im jest ona większa, tym łatwiej mechanizmy zabezpieczające wykrywają, że sieć lub konkretne urządzenie jest skanowane.

Topologia

Nmap posiada nakładkę oferującą GUI oraz kilka dodatkowych opcji, w tym ustalenie i zaprezentowanie w postaci grafu topologi skanowanej sieci.



Rysunek: Przykładowa topologia sieci (źródło: ▸ strona Zenmap)

Ruch sieciowy

Wizualizację topologii sieci oferował również program *Wireshark*, w czasach kiedy nazywał się EtherApe. Obecnie ta opcja nie jest dostępna, ale można go wykorzystać do innych celów, głównie do przechwytywania i badania pakietów sieciowych, nie tylko pochodzących z sieci Ethernet i WiFi, ale również Bluetooth. Dla sieci WiFi został opracowany program *Kismet*, który potrafi wykrywać identyfikatory SSID ukrytych sieci oraz wyposażony jest w narzędzia do deszyfrowania ruchu. Z kolei analizowanie, tworzenie i wysyłanie spreparowanych pakietów umożliwia np. narzędzie *hping3*.
Jeśli pentester ma dostęp do sieci wewnętrznej, to interesujące dla niego mogą być pakietu protokołu SNMP (ang. *Simple Network Management Protocol*). Pomocne informacje o narzędziach do skanowania sieci i nie tylko można znaleźć na stronie ▸ HighOn.Coffee.

Detekcja zabezpieczeń sieciowych

Istnieje wiele elementów sieciowych, które mogą wpłynąć na wyniki skanowania, a nawet przebieg całości testów penetracyjnych. Można je jednak wcześniej wykryć. Zaliczane do nich są:

- Rozwiązania typu **load balancer** — jeśli związane są z ruchem DNS lub HTTP, to mogą być wykryte przez narzędzie `lbd` (ang. *load balancing detector*). Bada ono różnice w nagłówkach i odpowiedziach serwera, żeby ustalić, czy zawsze pochodzą one z tego samego źródła. Czasem wystarczy jednak kilkukrotnie użyć narzędzi takich jak `nslookup` lub `ping`, by znaleźć zmianę adresów IP. Z kolei zapory dla aplikacji internetowych (WAF — *Web Application Firewall*) dodają lub zmieniają ciasteczka (ang. *cookies*) i nagłówki w odpowiedziach lub wysyłają pakiet FIN/RST by zakończyć niechciane połączenia.

Detekcja zabezpieczeń sieciowych

- Programy **antywirusowe** mogą być wykryte przez narzędzie *BeEF*, które służy do badania i eksploatacji podatności w przeglądarkach WWW.
- **Zapory sieciowe** — pomocne w ich wykrywaniu może być polecenie `traceroute` lub pokrewne. Również Nmap posiada skrypty, które pozwalają mu wykrywać „zwykłe” zapory, jak i te przeznaczone dla aplikacji internetowych. Po wykryciu zapory można sprawdzić które protokoły ona przepuszcza przy pomocy narzędzia *Firewalk*.

Inne zasoby

W zależności od ustalonego zakresu testów penetracyjnych konieczne może być zbadanie innych zasobów dostępnych w sieci badanego podmiotu. Do nich można zaliczyć *strony internetowe*, które mogą wymagać użycia narzędzi przeprowadzających operacje typu *crawling* i *scraping*, a nawet manualnego zbadania plików `robots.txt`. Innymi interesującymi zasobami mogą okazać się udziały sieciowe (Samba) lub usługi typu Active Directory oraz Kerberos. W ich przypadku istnieją zarówno gotowe rozwiązania, np. skrypty Nmap, jak i możliwość opracowania własnych narzędzi, np. w języku Python, którym posiada gotowe biblioteki do obsługi koniecznych protokołów. Zasoby dostarczone przez systemy chmurowe mogą być skanowane przy użyciu narzędzia ▸ CloudBrute. Dla dalszych działań w testach penetracyjnych może być przydatne pozyskanie tokenów uwierzytelniających (JWT, protokołu NTML, itp.). Przeprowadzając takie badania należy jednak pamiętać o ich zakresie.

Unikanie wykrycia

W przeciwieństwie do metod pasywnych, metody aktywne są narażone na wykrycie. Konieczność ukrycia ich stosowania jest różna w zależności od ustalonej metody testów penetracyjnych. W przypadku znanego środowiska może się ona pojawić, jeśli celem testów jest detekcja utajnionych (ang. *stealthy*) ataków. W przypadku nieznanego środowiska pojawia się ona niemalże zawsze, jednak z różnym nasileniem. Jeśli aktywne gromadzenie informacji przeprowadzane jest z zewnątrz sieci, to może zostać zignorowane przez osoby odpowiedzialne za bezpieczeństwo badanego systemu, z powodu natłoku innych podobnych działań. Jednakże przeprowadzenie skanowania z wnętrza sieci jest już bardziej narażone na wykrycie. Narzędzia służące do tej czynności mają szereg opcji, które pozwalają ją uczynić trudniej wykrywalną, takich jak spowolnienie działania, losowanie portów, skanowanie rozproszone, fałszowanie adresów IP.

Obrona

Celem testów penetracyjnych jest wskazanie słabych punktów weryfikowanego systemu, ale również wskazanie potencjalnych sposobów ich usunięcia. Dotyczy to również zapobiegania wyciekowi informacji, które mogą posłużyć do przełamania zabezpieczeń.
Aby utrudnić stosowanie aktywnych metod rekonesansu można:

- ograniczyć dostępność usług na zewnątrz tylko do tych niezbędnych;
- użyć rozwiązań IPS lub podobnych, które ograniczają lub uniemożliwiają próby skanowania;
- wdrożyć systemy monitorowania i powiadamiania.

Okazuje się, że przeciwdziałanie metodom pasywnym jest bardziej skomplikowane. Przede wszystkim warto opracować politykę przekazywania informacji na zewnątrz, a także wdrożyć szkolenia, które uczulą personel na problem nawet nieświadomego wycieku danych.

Obrona

Pasywnym metodom związanym z odpytywaniem usług DNS można zapobiegać poprzez:

- blokowanie (ang. *blacklisting*) systemów i sieci, które nadużywają tych usług,
- zastosowanie rozwiązań typu CAPTCHA celem zablokowania botów,
- skorzystanie z usług zewnętrznych firm, które rejestrują domeny w imieniu innych podmiotów,
- ograniczenie liczby zapytań w ciągu ustalonego czasu,
- niepublikowanie plików z danymi strefy DNS, jeśli nie jest to konieczne.

Pytania

?

Koniec

Dziękuję Państwu za uwagę!