Wykrywanie ataków w sieciach komputerowych z wykorzystaniem systemów Honeypot

Maciej Jagiełło

Streszczenie Artykuł opisuje projekt architektury sieci honeypotów przeznaczonej do wykrywania ataków sieciach komputerowych. Opisane są popularne honeypoty użyte w systemie. Przedstawione są narzędzia do wdrażania systemu. Zamieszczony jest krótki opis aplikacji oraz pokazane są bliźniacze projekty.

Słowa kluczowe: honeypot, glastopf, kippo, dionaea, vagrant, puppet, virtualbox

1 Wprowadzenie

Wykrywanie ataków w sieciach komputerowych to ważny element działań na rzecz zwiększania bezpieczeństwa systemów. Cyberprzestępczość rozwija się szybkim tempem. Z każdym dniem opracowywane są nowe sposoby ataków, wykorzystywane są nowe podatności w aplikacjach, protokołach i systemach operacyjnych. Aby temu przeciwdziałać, konieczne jest opracowanie odpowiednich narzędzi.

Celem systemu jest stworzenie narzędzia do zbierania adresów IP[1] hakerów, botów[2] i zainfekowanych maszyn[3] w spójną bazę danych[4], którą będzie można później wykorzystać np. do rozszerzania reguł zapór ogniowych[5]. System ma być też źródłem nowych wirusów do analizowania przez antywirusy. Wybrane rozwiązanie problemu to sieć Honeypotów[6].

Honeypot to jedynie idea zbierania informacji. Jest to kombinacja ustawień sprzętowych i programowych, które tworzą system pułapkę. Ustawienia sprzętowe, czyli komputer, wyodrębniony obszar sieci lokalnej, który odpowiednio symuluje prawdziwą sieć, ale jest od niej odizolowany i zabezpieczony.

Taki system, uruchomiony na maszynie serwerowej[7], udostępnia w Internecie jakiś zasób atrakcyjny dla atakującego. Może to być rekord w bazie danych,

[1] To taki adres np. 127.0.0.1, który identyfikuje komputer w sieci.
[2] Programy, które wykonują coś automatycznie. Najczęściej są instalowane przez hakerów jako wirusy, ale mogą być też uruchamiane z jego osobistego komputera.
[3] Bardziej ogólna nazwa dla komputera
[4] Taka baza danych to najczęściej tabela wyglądająca jak w Excelu.
[5] ang. firewall - zatrzymuje niechciane połączenia do komputera. Jedna z prostszych metod przeciwdziałania atakom. Domyślnie instalowany w Windowsach
[6] ang. Garnek z miodem. Taka atrapa do łapania hakerów
[7] Maszyna serwerowa to po prostu komputer

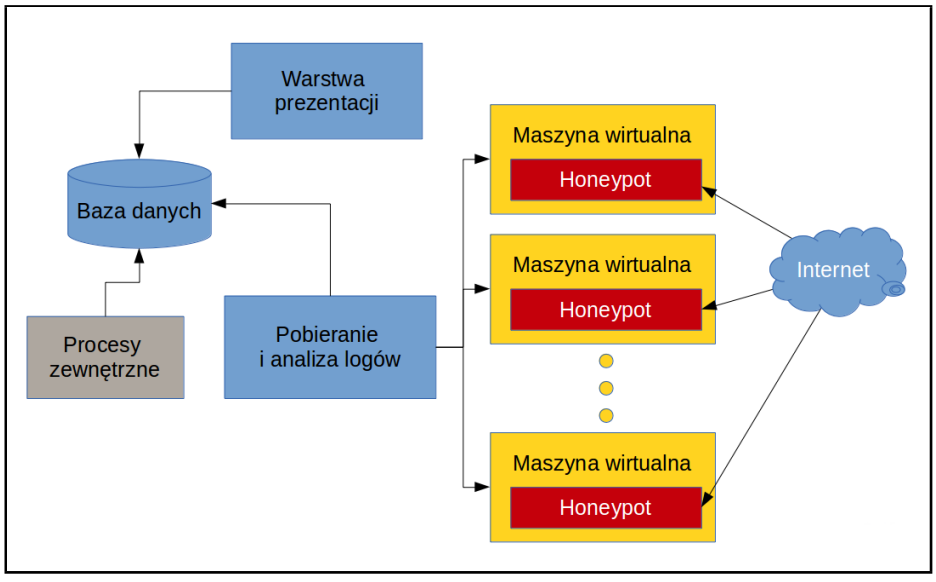
aplikacja lub cały system operacyjny. Używając odpowiednich mechanizmów do monitorowania zachowania, zbiera dane o sposobach wykonywania ataków.

Jego główną zaletą jest to, że korzystają z niego niemal wyłącznie użytkownicy, którzy chcą włamać się do sieci.

Aby z sukcesem doprowadzić do wdrożenia i utrzymania takiego systemu niezbędne są technologie wspomagające wdrożenie[8]. Przedstawione zostaną one w kolejnych rozdziałach artykułu. Działający system, który dostarcza już dane, będzie mógł je prezentować w formie przyjaznej dla człowieka. Ta część systemu przedstawiona zostanie na przykładzie podobnych projektów.

2 Sieć honeypotów

Na rysunku 1 przedstawiony jest schemat architektury aplikacji. Każdy honeypot uruchomiony jest w maszynie wirtualnej[9]. Maszyna ta posiada dostęp do Internetu i wystawia do niego, odpowiednie dla zainstalowanych honeypotów, zasoby. Moduł pobierania i analizy logów zbiera pozostawione przez honeypoty informacje o atakujących. Są to ślady wykonywanych akcji i pliki binarne[10] ładowane na maszyny przez atakujących.



Rysunek 1: Architektura aplikacji.

[8] Wdrożenie polega na instalacji i konfiguracji aplikacji gdzieś na jakimś serwerze

[9] Maszyna wirtualna to system operacyjny w systemie operacyjnym.

[10] Pliki binarne (przeciwieństwo to pliki tekstowe) to pliki, których się nie czyta, bo są skompresowane i reprezentowane jako bajty, a nie jako litery

Istnieje bardzo dużo realizacji honeypotów. Na stronie [2], która zajmuje się rozwojem narzędzi do zwiększenia bezpieczeństwa w Internecie, jest wymienione około 30 projektów, a to z pewnością nie wszystkie dostępne.

Warto wspomnieć o pracującym na wydziale Elektroniki i Technik Informacyjnych[11], doktorze Krzysztofie Cabaju, który na przedmiotach BSS, SKM2 i TKOM prowadzi studentów podczas implementacji[12] autorskich honeypotów.

W opisywanym systemie zostały wybrane trzy realizacje honeypotów ze względu na wystarczającą różnorodność w oferowanych przez nie usługach.

3 Honeypoty

3.1 Glastopf

Pierwszy z prezentowanych honeypotów, Glastopf[13], jest zwykłą stroną www, którą można znaleźć w wyszukiwarkach internetowych. Treść strony zawiera frazy w języku angielskim, które są generowane z dostarczonej przez twórców bazy danych. Glastopf potrafi w prosty sposób analizować zapytania HTTP[14], w szczególności te zapytania GET, które w adresie URL zawierają inne adresy URL do zewnętrznych plików. Takie argumenty parsuje[15] i próbuje ściągać na dysk do późniejszej analizy.

3.2 Kippo

Kolejnym wektorem ataków, na jaki są podatne komputery w sieciach komputerowych jest protokół[16] ssh[17]. Kippo symuluje sesję SSH udostępniając prostego shella[18]. Najczęściej używane polecenia mają napisane własne implementacje, które oszukują użytkownika tego środowiska[19]. Polecenia rzadziej używane zwracają błąd, w taki sposób, aby nie zdradzić fałszywości systemy. Kippo udostępnia dla nich fałszywy system plików[20] tylko do odczytu, który wygląda jak po świeżej instalacji Debiana[21].

Ta aplikacja loguje informacje o adresach IP, próbach pobrania plików z Internetu, zapisuje te pliki w kwarantannie, oraz loguje całą sesję użytkownika

[11] Mój wydział
[12] Implementacja to popularne słowo na pisanie aplikacji
[13] niem. Szklany garnek
[14] Gdy wejdziesz na stronę google.pl, to tak naprawdę wyślesz zapytanie GET protokołem HTTP do serwera o domenie google.pl.
[15] Analizuje jego znaczenie. Tekst dla komputera jest niezrozumiały póki nie będzie wiedzieć jak go czytać. Parsowanie to proces, który zamienia jeden ciąg literek w inne struktury danych
[16] Protokołem do stron internetowych jest HTTP.
[17] Protokół do zdalnego wykonywania komend w konsoli na komputerze
[18] Aplikacja do uruchamiania komend
[19] Atakujący musi myśleć, że jest na prawdziwym komputerze, a nie na atrapie
[20] To co zaczyna się od C: w Windowsie
[21] System operacyjny typu Linux

od próby zalogowania na serwer, aż do aktywności po wylogowaniu. Robi to przechwytując polecenia „logout“, „exit“ lub znak wysyłany kombinacją klawiszy Ctrl + D. Następnie wyświetla znak zachęty[22] wyglądający jak na lokalnym komputerze.

3.3 Dionaea

Ostatnim z użytych w systemie honeypotów jest Dionaea. Protokoły, którymi się posługuje do serwowania pułapki to głównie Samba, HTTP, FTP, MySQL, RTP. Samba jest bardzo popularnym protokołem do wymiany plików w Windowsie. Wiele osób bez zaawansowanej wiedzy o tym systemie korzysta z niego i nieumyślnie otwiera sobie drogę do włamania. Dlatego atakujący często szukają drogi właśnie tym kanałem.

4 Środowisko uruchomieniowe

Istotnym, choć niekoniecznym krokiem w stronę bezpieczeństwa systemu jest uruchamianie serwerów honeypotowych w maszynie wirtualnej. Pozwala to osiągnąć wysoki poziom separacji środowiska podatnego na ataki, od rzeczywistej maszyny, nad którą nie można stracić dostępu. Dodatkowo to rozwiązanie jest bardzo przydatne gdy jest potrzeba, by system był łatwy w replikacji na inne maszyny. Aplikacja uruchamiająca maszynę wirtualną jest odpowiedzialna za dostarczenie takiego samego środowiska. W opisywanym systemie tą aplikacją jest VirtualBox rozwijany przez firmę Oracle.

Dodatkową warstwą nad VirtualBoxem jest Vagrant. Jest to świetne narzędzie do automatyzacji instalacji maszyny wirtualnej z użyciem skryptów konfiguracyjnych[23]. Jego użycie polega na przygotowaniu skryptu instalacyjnego[24] na naszej maszynie. W tym systemie do instalacji honeypotów został użyty Puppet. Puppet to narzędzie do deklaratywnego[25] zarządzania stanem[26] maszyny.

Alternatywą do wirtualnej maszyny jest wirtualny kontener[27], który nie tworzy wirtualnego systemu od początku do końca, a tylko izoluje aplikację od systemu. Najpopularniejszym rozwiązaniem jest Docker. Docker potrzebuje mniej zasobów systemowych[28], ale nie gwarantuje 100% bezpieczeństwa.

---

[22] Znak, który oznacza, że konsola oczekuje kolejnego polecenia
[23] Skrypt konfiguracyjny to jakaś aplikacja
[24] Znów aplikacja
[25] Deklaratywny sposób to pisanie jaki ma być stan, czyli CO ma być zrobione. Np. woda ma być ugotowana. Przeciwieństwo - imperatywny sposób - to pisanie JAK ma coś być zrobione np. wstaw wodę i zacznij gotować aż osiągnie 100 st. C.
[26] Zainstalowanymi aplikacjami, stworzonymi plikami
[27] Sam nie wiem co to jest :)
[28] Procesor, Dysk, Ram

4.1 Narzędzia

Wartym do wejścia w szczegóły tematem jest wspólne działanie Vagranta, VirtualBoxa i Puppeta. Schemat jest przedstawiony na rysunku 2. Numery obok strzałek oznaczają kolejne kroki instalacji honeypota na maszynie wirtualnej.

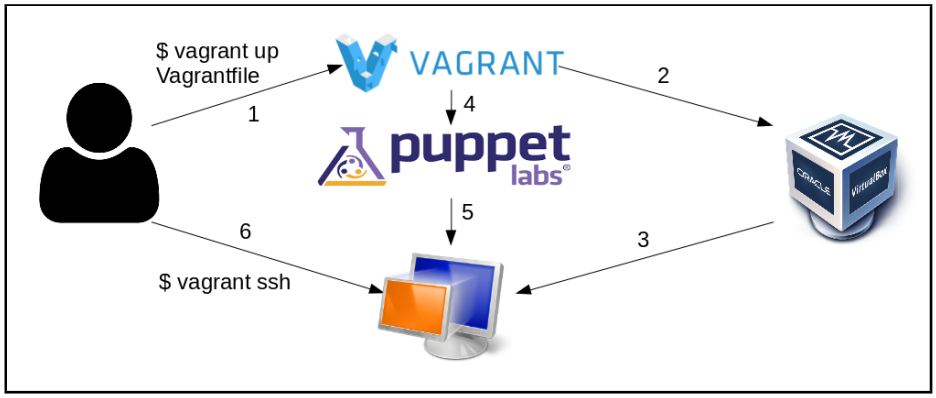
Vagrant i VirtualBox

Zanim można rozpocząć procedurę, należy pobrać dystrybucję[29] systemu poleceniem:

```
1 $ vagrant init hashicorp/precise32
```

Następnie polecenie `vagrant up` uruchamia procedurę. Najpierw Vagrant, korzystając z VirtualBoxa tworzy wirtualne środowisko. Potem uruchamia skrypty Puppetowe. One instalują się na maszynie tworząc gotowe środowisko.

Dostęp do stworzonej maszyny można uzyskać poprzez polecenie `vagrant ssh`.



Rysunek 2: Vagrant, Puppet, VirtualBox.

Cały proces trwa około 5 minut, został wykonany z użyciem 3 poleceń w terminalu[30], a wynikiem jest maszyna wirtualna z ubuntu server w 32 bitowej wersji z zainstalowanym honeypotem.

Puppet

Przykładowy i skrócony na potrzeby artykułu plik puppetowy jest na listingu 1.1.

[29] Konkretną wersję systemu operacyjnego. Np. wspomniany Debian
[30] Inaczej konsola

```
Vcsrepo['/opt/BFR'] -> Exec['install_bfr']
Service['apache2'] -> Package['glastopf']

vcsrepo { '/opt/BFR':
  ensure   => present,
  provider => git,
  source   => 'git://github.com/glastopf/BFR.git',
}

exec { 'install_bfr':
  command => 'make && make install; echo "zend=bfr.so" >> /
    etc/php5/cli/php.ini',
  cwd     => '/opt/BFR',
  unless  => 'grep -q "zend=bfr.so" /etc/php5/cli/php.ini',
}

service { 'apache2':
  ensure => stopped,
  enable => false,
}

package { 'glastopf':
  ensure   => present,
  name     => 'glastopf',
  provider => 'pip',
}
```

Listing 1.1: glastopf.pp

W skrypcie puppetowym w sposób deklaratywny określa się jak ma wyglądać stan maszyny po instalacji. Istnieje wiele wbudowanych modułów[31], które pozwalają zarządzać plikami, serwisami, aplikacjami, skryptami do uruchomienia, itp. Do zbioru modułów można doinstalowywać te udostępnione przez społeczność. A w nich są takie, które np. zarządzają gitem[32].

Do uruchomienia skryptu z listingu 1.1 potrzebujemy tylko jednego polecenia:

```
$ puppet apply glastopf.pp
```

Podczas wykonywania tego polecenia, Puppet analizuje plik `pp` i interpretuje go z użyciem modułów. Moduły wbudowane są napisane w języku Ruby i mogą być przygotowane dla różnych systemów operacyjnych. To, który się wykona zależy od środowiska, które rozpozna Puppet.

[31] Bibliotek, które tak naprawdę są kolejnymi aplikacjami

[32] Aplikacja do śledzenia zmian w plikach podczas pisania programów, tak, żeby jak coś się przypadkiem usunie, żeby dało się do tego wrócić.

5 Aplikacja

Silnik do zbierania danych to aplikacja napisana w języku Java. Jego działanie polega na logowaniu się na serwer za pomocą protokołu SSH i pobraniu logów[33]. To gdzie się one znajdują jest zdefiniowane dla każdego honeypota oddzielnie. Na metadane[34] o każdym z honeypotów składają się:

- lista lokalizacji plików do ściągnięcia,
- reguły rozpoznawania plików tekstowych.

Aplikacja archiwizuje i utrwala pobrane dane w bazie danych. Analizuje ona również aktywność atakujących wg reguł zdefiniowany w metadanych, tak aby wyciągnąć adres IP komputera wysyłającego atak, stempel czasowy ataku i jego typ. Dane te później agreguje i udostępnia w formie zrozumiałej dla warstwy prezentacji, tak jak na rysunku **??**.

Jednym z problemów przy pobieraniu logów jest to, że ze względów bezpieczeństwa honeypot nie powinien mieć dostępu do żadnej części tworzonego systemu, ponieważ w przypadku włamania się na serwer, haker może zniszczyć nie tylko maszynę z honeypotem.

Zostało to rozwiązane właśnie używając protokołu SSH. Nie wymaga on instalowania dodatkowych aplikacji, wspiera przesyłanie plików i jest protokołem względnie bezpiecznym w porównaniu do innych autorskich rozwiązań.

Kolejnym problemem, który można spotkać podczas używania i konserwacji systemu jest rosnący bez końca plik z logami. Aby sobie z tym poradzić należy go okresowo archiwizować i usuwać, jeśli przekroczy określony rozmiar. Użyta została do tego aplikacja dostępna w repozytorium Ubuntu - logrotate.

Testy integracyjne aplikacji zostały wsparte przez technologie użyte przy wdrażaniu honeypotów: Vagrant, Virtualbox i Puppet.

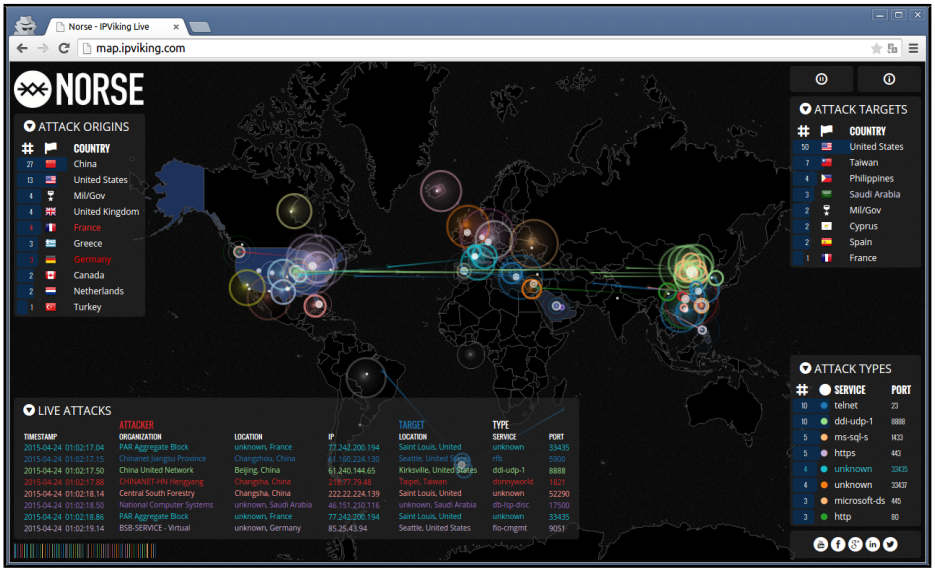
6 Warstwa prezentacji

Istnieje kilka systemów dostępnych w Internecie, którym dane do prezentacji mógłby dostarczać system przedstawiony w artykule.

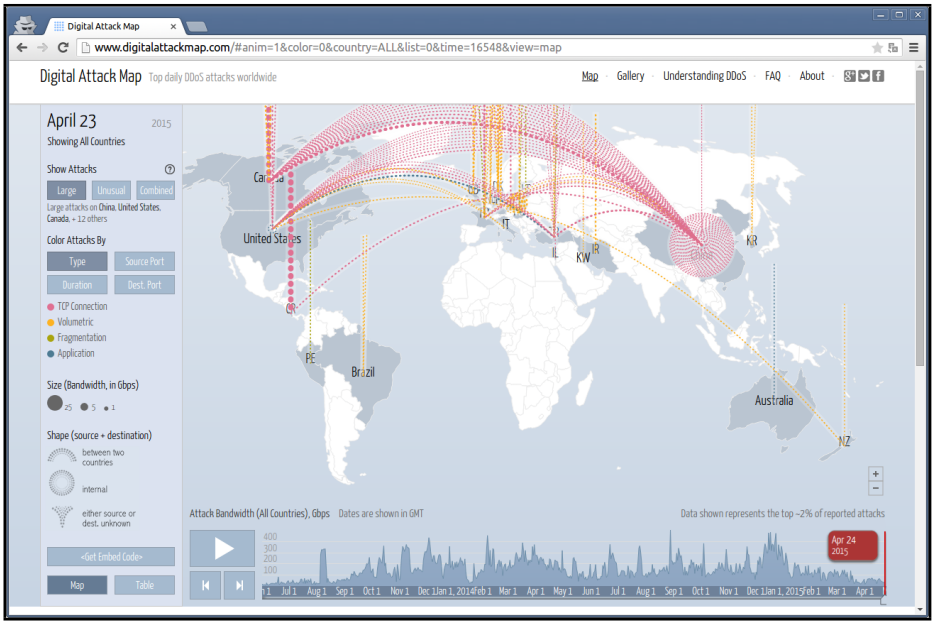
Pierwszym z nich jest Norse - IPViking Live, którego wygląd pokazany jest na rysunku 3. Przedstawia on mapę świata, na której wyświetlane są ataki na żywo. Jego atrakcyjność zwiększona jest przez dynamikę aktualnych ataków, niestety niemożliwej do przedstawienia na jednym rysunku.

Nie istnieje wiele kluczowych różnic od drugiej mapy ataków, pokazanej na rysunku 4, którą powołał do życia Google ideas, Big Picture Group, a dane dostarcza Arbor Networks. Z pewnością niewiele danych jest wspólnych między tymi dwoma projektami. Skala ataków jest na tyle duża, że nie sposób tego kontrolować w sposób pełny.

[33] Plików tekstowych z wpisami, które zostawił honeypot
[34] Metadane to dane o danych



Rysunek 3: Norse – IPViking Live.



Rysunek 4: Digital Attack Map.

7 Podsumowanie

Do wykonania systemu wykrywającego ataki w sieciach komputerowych został użyty system honeypotów. Wdrażany przy użyciu technologii Vagrant, VirtualBox, Puppet i Java. Użyte honeypoty to Glastopf, Dionaea i Kippo, które oczekiwały na atakujący z użyciem protokołów HTTP, SSH, FTP, MySQL i RTP.

Pierwsze dane od botów skanujących internet pojawiły się już po paru dniach. Po kilku tygodniach średnia liczba ataków na jeden honeypot to 50 ataków na godzinę.

Boty atakujące serwer Glastopf najczęściej posiadały w nagłówku zapytania[35] informację, że to tylko zbieranie danych na potrzeby badań edukacyjnych. Jednak rzeczywistość każe sądzić, że to tylko próba odwrócenia uwagi od faktycznego ataku.

Literatura

1. http://jbeczala.republika.pl/files/ataki_sieciowe.pdf
2. https://www.honeynet.org/project
3. http://glastopf.org/
4. https://github.com/desaster/kippo
5. https://www.honeynet.org/project
6. http://map.ipviking.com/
7. http://digitalattackmap.com/
8. https://puppetlabs.com/
9. https://www.virtualbox.org/
10. https://www.vagrantup.com/
11. http://www.digitalforreallife.com/2012/11/boosting-teamwork-with-vagrant/

[35] Np. przeglądarki internetowe w nagłówku wysyłają to jaką są przeglądarką, jaka jest ich wersja, jaka jest wersja systemu operacyjnego