Wyznaczanie rozkładu geograficznego potencjalnego zagrożenia, w oparciu o zebrane zablokowane adresy IP z pliku hosts.deny

Dydaktyka Informatyki 11, 161-166

2016

Dydaktyka Informatyki 11(2016)
ISSN 2083-3156
DOI: 10.15584/di.2016.11.21
http://www.di.univ.rzeszow.pl

Wydział
Matematyczno-Przyrodniczy UR
Laboratorium Zagadnień
Społeczeństwa Informacyjnego

Jacek WOŁOSZYN

Dr inż., Uniwersytet Technologiczno-Humanistyczny w Radomiu, Wydział Informatyki i Matematyki, Katedra Informatyki, ul. Malczewskiego 29, 26-600 Radom; jacek@delta.pl

WYZNACZANIE ROZKŁADU GEOGRAFICZNEGO POTENCJALNEGO ZAGROŻENIA W OPARCIU O ZEBRANE ZABLOKOWANE ADRESY IP Z PLIKU HOSTS.DENY

THE GEOGRAPHICAL DISTRIBUTION OF THE POTENTIAL THREAT BASED ON THE COLLECTED BLOCKED IP ADDRESSES FROM FILE HOSTS.DENY

Słowa kluczowe: system operacyjny, dzienniki, przetwarzanie potokowe.
Keywords: system security, firewall, identification, encryption.

Streszczenie

Rozwiązania serwerowe oparte na systemach linuxowych pozwalają zablokować zdalny dostęp do swoich zasobów poprzez wpisanie adresu IP do pliku /etc/hosts.deny. Jest to skuteczna metoda pozwalająca na blokadę szczególnie uciążliwych klientów, których zamiary są bliżej nieokreślone. W tym artykule przedstawiono przykład zamiany pozyskanych adresów IP na powiązane z nimi położenie geograficzne, celem uzyskania rozkładu geograficznego potencjalnego zagrożenia.

Summary

Server solutions based on systems Linux boxes allow to block remote access to their resources by typing the IP address into the /etc/hosts.deny file. This is an effective method of blocking a particularly onerous customers whose intentions are vague. This article is an example of the conversion of obtained IP addresses to related geographical locations to obtain the geographical distribution of potential danger.

Wstęp

Systemy serwerowe udostępniające usługi sieciowe w szerokopasmowym Internecie są szczególnie narażone na zagrożenia. Większość narzędzi zabezpieczających hosta przed skanowaniem zapisuje wyniki swojej pracy dodając 'zbyt

aktywne' adresy IP do pliku hosts.deny. uniemożliwiając im tym samym dalszą komunikację. Analiza tego pliku pokazuje skalę problemu. Okazuje się bowiem, że dzienny przyrost zablokowanych adresów IP to 40–300 dziennie. Okazuje się zatem, że każdego dnia mamy do 300 prób nieautoryzowanych wejść do systemu. Artykuł przedstawia próbę korelacji zablokowanych adresów IP z ich rozkładem geograficznym. Przykład opisany w niniejszym artykule został przedstawiony w oparciu o rzeczywiste dane uzyskane z systemu serwerowego udostępniającego zasoby www dla wybranej grupy użytkowników. W systemie działa również usługa ssh pozwalająca na zdalną pracę osób zajmujących się administrowaniem systemu. To właśnie ona powoduje ogromne zainteresowanie nieautoryzowanych klientów.

1. Zawartość pliku hosts.deny

Plik hosts.deny po instalacji systemu nie zawiera żadnych zapisów. W celu zablokowania transmisji pomiędzy serwerem a wybranym klientem serwera, należy wpisać jego adres do pliku hosts.deny. Takie działanie jest jak najbardziej dobrym rozwiązaniem w przypadku pracy systemu w sieci firmowej, korporacyjnej, gdzie struktury sieci są dokładnie opisane i znany jest rozkład przydziału poszczególnych usług do klientów. Jednak w tym przypadku mamy do czynienia z siecią otwartą, gdzie użytkownicy nie pracują pod określonymi adresami IP. W takiej sytuacji nie można dokładnie sprecyzować, które IP są uprawnione do nawiązywania transmisji. Analizując logi np. auth.log jak na listingu 1, widać zapisy tych, którzy usilnie próbują dostać się do systemu wykorzystując różne metody, np. brute force. To jest dla nas informacja, że takim adresom należy zablokować transmisję.

```
Jan 16 14:18:46 dot sshd[29891]: refused connect from 176.111.36.26 (176.111.36.26)
Jan 16 14:22:43 dot sshd[29903]: warning: /etc/hosts.allow, line 13: host name/name mismatch:
    212.156.88.46.static.turktelekom.com.tr != www.turktelekom.com.tr
Jan 16 14:22:44 dot sshd[29903]: refused connect from 212.156.88.46 (212.156.88.46)
Jan 16 14:22:44 dot sshd[29904]: warning: /etc/hosts.allow, line 13: host name/name mismatch:
    212.156.88.46.static.turktelekom.com.tr != www.turktelekom.com.tr
Jan 16 14:22:44 dot sshd[29905]: warning: /etc/hosts.allow, line 13: host name/name mismatch:
    212.156.88.46.static.turktelekom.com.tr != www.turktelekom.com.tr
Jan 16 14:22:44 dot sshd[29904]: refused connect from 212.156.88.46 (212.156.88.46)
```

Listing 1. Fragment pliku auth.log

Wyżej zamieszczony listing pokazuje niewielki wycinek informacji uzyskany z zapisów z logami. Nie jest w tym przypadku istotny charakter wpisanej informacji, a jedynie numer IP próby nawiązania połączeń. Po takiej analizie można samodzielnie umieścić numer IP w pliku hosts.deny, ale w przypadku systemu pracującego w rzeczywistej sieci jest to uciążliwe ze względu na ogromną liczbę takich incydentów.

2. Opis problemu

W artykule[1] przedstawiono aplikację Portsentry monitorującą porty w systemie operacyjnym i reagującym na próbę podejrzanej aktywności klienta zdalnego. Jeżeli działanie maszyny zdalnej nie ogranicza się do typowych zachowań, program podejmuje działania określone w pliku konfiguracyjnym i jako rezultat swoich działań umieszcza adres IP maszyny w pliku hosts.deny zabraniając mu komunikacji. Jest to typowa reakcja na działanie w przypadku pingowania, skanowania IP i portów przeważnie z wykorzystaniem programu nmap, próby wejścia do systemu z wykorzystaniem ataków słownikowych itp. Podobną funkcjonalnością charakteryzuje się aplikacja denyhosts. W wyniku działań tych aplikacji w treści pliku /etc/hosts.deny umieszczane są adresy hostów, które wykazują zachowanie wskazujące na próbę nieautoryzowanego połączenia.

Może nie byłoby nic nadzwyczajnego w tym działaniu, ale okazało się, że baza dopisywanych numerów IP powiększała się bardzo szybko. W ciągu roku działania systemu aplikacje dopisały do pliku ok 108 124 numerów IP, które aplikacje uznały za próbę ataku.

Informacje o ilości zapisanych wierszy można uzyskać wydając polecenie[2]:

```
cat /etc/hosts.deny | wc -l
```

Rozmiar pliku to ponad 2MB tylko za zapisami numerów IP.

Kim są osoby kryjące się za zablokowanymi adresami IP, z jakich pochodzą części świata, z jakich krajów, z jakich miast. Jakimi kierują się przesłankami tak postępując. Problem jest na pewno ciekawy i należałoby poświęcić temu oddzielny artykuł. W tym artykule ograniczymy się do opisu technicznej części relacji pomiędzy adresami IP a powiązaniem geograficznym.

3. Rozwiązanie problemu

Zgromadzone przez opisane wcześniej aplikacje dane w postaci adresów IP skutecznie uniemożliwiają komunikację pomiędzy nimi a serwerem i doraźnie rozwiązują problem związany z bezpieczeństwem, jednak chcielibyśmy się dowiedzieć więcej o zapisanych adresach IP. Chociażby z jakiego regionu geograficznego pochodzą. Być może analiza problemu z użyciem takich danych pozwoliłaby uzyskać odpowiedź, co stoi za tak intensywną próbą chęci dostania się do zasobów serwera.

---

[1] J. Wołoszyn, *Wykorzystanie aplikacji Portsentry do aktywnej ochrony systemu serwerowego opartego na systemie Linux* [w:] *Dnesne Trendi Innovacii*, red. L. Varkoly, DTI v Dubnićy 2013.

[2] N. Gift, J. Jones, *Python for Unix and Linux system Administration*, O'Reilly 2008.

Aby uzyskać informację o przypisaniu adresu IP w sposób tradycyjny wystarczy wydać polecenie whois IP.

Wydając polecenie whois i podając adres IP z listingu 1 212.156.88.46 otrzymamy informację:

```
% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
%       To receive output for a database update, use the "-B" flag.

% Information related to '212.156.0.0 - 212.156.158.255'

% Abuse contact for '212.156.0.0 - 212.156.158.255' is 'abuse@ttnet.com.tr'

inetnum:        212.156.0.0 - 212.156.158.255
netname:        TTNET
descr:          Turk Telekom TTnet national backbone
country:        TR
admin-c:        TTBA1-RIPE
tech-c:         TTBA1-RIPE
status:         ASSIGNED PA
mnt-by:         AS9121-MNT
created:        2007-03-21T12:07:43Z
last-modified:  2007-03-21T12:07:43Z
source:         RIPE # Filtered

role:           TT Administrative Contact Role
address:        Turk Telekom Genel Mudurlugu
phone:          +90 312 555 1920
fax-no:         +90 312 313 1924
admin-c:        BADB3-RIPE
abuse-mailbox:  abuse@ttnet.com.tr
tech-c:         BADB3-RIPE
tech-c:         BADB3-RIPE
tech-c:         BADB3-RIPE
nic-hdl:        TTBA1-RIPE
mnt-by:         AS9121-MNT
created:        2002-02-28T12:22:28Z
last-modified:  2015-12-31T12:23:35Z
source:         RIPE # Filtered

% Information related to '212.156.64.0/19AS9121'
route:          212.156.64.0/19
descr:          TurkTelekom
```

```
origin:         AS9121
mnt-by:         AS9121-MNT
created:        2011-05-25T14:04:14Z
last-modified:  2011-05-25T14:04:14Z
source:         RIPE # Filtered

% This query was served by the RIPE Database Query Service version 1.83.1 (DB-4)
```

Listing 2. Informacje o adresie IP uzyskane w sposób tradycyjny

Otrzymane dane z bazy dość szczegółowo określają przynależność adresu. Jednak nie wszystkie informacje są nam potrzebne i co ważniejsze ręczne odpytywanie bazy w przypadku kilkudziesięciu lub kilkuset tysięcy adresów czynią zadanie niewykonalnym. Można polecenie umieścić w pętli i odpytywać kolejno z bazy każdy numer IP, jednak czy zdalny system umożliwi taką procedurę i przypadkiem nie podejmie decyzji, że jest to atak typu DOS?

Zdecydowanie lepszym rozwiązaniem na tym etapie jest skorzystanie z bazy adresów IP typu offline zamieszczonej na stronie: https://www.maxmind.com/en/geoip-demo i napisanie aplikacji w Pythonie[3] pozwalającej na szybkie zdekodowanie adresu IP do oczekiwanej przez nią postaci, czyli w tym przypadku nazwy kraju czy miasta.

Samo wykorzystanie tej bazy poza ściągnięciem jej na lokalny dysk do katalogu wymaga zaimportowania modułu pygeoip. Moduł można zainstalować wydając polecenie pip install pygeoip[4].

Kolejnym ważnym krokiem jest podpięcie do zmiennej pliku hosts.deny
f = open('/root/hosts.deny', 'rt') ,
jak i również wskazanie źródła pobranej bazy

```
gic = pygeoip.GeoIP('/usr/share/GeoIP/GeoLiteCity.dat')
for i in ip:
   if re.match(r'sshd', i): #dopasowanie wzorca powłoki

      vv = v.lstrip('sshd: ')
      if gic.record_by_addr(vv):
         poz = gic.record_by_addr(vv)
         if re.match(r'PL', poz['country_code']):
            print ('%10s %10s %10s' % (poz['city'],poz['country_code'],vv))
```

Opisany powyżej wycinek procedury pobiera z pliku hosts.deny wiersze, które zawierają wzorzec sshd:, a następnie dekodują pozostały adres IP na na-

[3] A. Downey, *Python for Software Design*, Cambridge University Press 2009; A. Downey, *Think Python*, O'Reilly 2012; M. Goodrich, R. Tamassia, M. Goldwasser, *Data Structures and Algorithms in Python*, Wiley 2013; D. Hellman, *The Python Standard Library by Example*, Addison-Wesley 2011; Y. Hilpisch, *Derivatives Analytics with Python*, Wiley 2015.

[4] J. Payne, *Beginning Python*, Wrox 2010; M. Summerfield, *Programming in Python 3*, Addison-Wesley 2010; T. Ziade, *Packt, Expert Python Programming*, Publishing 2008.

zwę miasta. Z bazy adresów w przedstawionym przykładzie wybierane są tylko adresy mające w zapisie kod PL przez, zastosowanie w pętli kolejnego wzorca z kodem kraju w tym przypadku PL, przez co wypisywane są tylko miejscowości z Polski.

```
Cracow       PL 83.30.91.19
Bydgoszcz    PL 94.141.149.11
Przemysl     PL 195.117.119.210
Wschowa      PL 83.2.52.99
Lodz         PL 85.89.188.194
Uzdowo       PL 83.28.202.13
Kalisz       PL 77.89.76.114
```

Wnioski

Opisany przykład przedstawia ogólne podejście do problematyki. Grupując zebrane wyniki można przedstawić je w postaci rozkładu według kraju, miasta, kontynentu czy innych danych udostępnianych przez pola bazy. Takie pogrupowanie jest wstępem do bardziej zaawansowanej analizy uzyskanych danych. Zestawienie uzyskanych rozkładów z innymi wynikami badań może dać interesujące wyniki dające odpowiedź na pytanie, co kieruje osobami kryjącymi się pod tymi adresami do takich zachowań, czy jest to tylko ich pasja, a może praca na czyjeś zlecenie. Czy większa aktywność osób z wybranego kraju jest tylko związana z jego populacją, a może są inne źródła takiej działalności.

Takie rozważania zostaną podjęte w przyszłości w kolejnych artykułach. Na tym etapie pokazano jedynie mechanizm szybkiego pozyskiwania informacji z zebranych danych.

Bibliografia

Downey A., *Python for Software Design*, Cambridge University Press 2009.
Downey A., *Think Python*, O'Reilly 2012.
Gift N., Jones J., *Python for Unix and Linux system Administration*, O'Reilly 2008.
Goodrich M., Tamassia R., Goldwasser M., *Data Structures and Algorithms in Python*, Wiley 2013.
Hellman D., *The Python Standard Library by Example*, Addison-Wesley 2011.
Hilpisch Y., *Derivatives Analytics with Python*, Wiley 2015.
Payne J., *Beginning Python*, Wrox 2010.
Summerfield M., *Programming in Python 3*, Addison-Wesley 2010.
Wołoszyn J., *Wykorzystanie aplikacji Portsentry do aktywnej ochrony systemu serwerowego opartego na systemie Linux*, Dnesne Trendi Innovacii, Varkoly DTI 2013, ISBN 978-80-89400-60-7.
Ziade T., *Packt, Expert Python Programming*, Publishing 2008.