УДК 681.32

Фучило (Бабій) Ю.С. [1]***, Кузик В.М.*** [2]***, Раїнчук В. В.*** [1]
[1]*Тернопільський національний економічний університет*
[2]*Галицький коледж ім. В. Чорновола*

СИСТЕМА ФІЛЬТРАЦІЇ КОНТЕНТУ НА БАЗІ CISCO І FREEBSD

Вступ. При побудові захищених корпоративних мереж, як і використанні інтернет речей, важливою задачею є фільтрування вхідного та вихідного трафіку[1]. Одним з складних видів такої фільтрації є фільтрація контенту, що проходить мережевими каналами. Існує безліч систем фільтрації контенту проте вони усі мають складені та персоналізовані налаштування під конкретну установу.

Мета: Розробка узагальненої схеми системи фільтрації мережевого трафіку за контентом є метою даного дослідження.

1. Розробка пристрою

Досить поширеним з варіантів фільтрації трафіку є використання SQUID на базі FreeBSD, зістикований з бекбону маршрутизатором CISCO по протоколу WCCP2.

При дослідженні методів фільтрації[2] та аналізу продуктивності подібних систем виявлено, що якщо пропустити в фільтр весь http-трафік, то фільтрація повинна здійснюватися на швидкостях понад гігабіта в секунду. Для такого завдання було розроблено наступну схему(Рисунок 1).

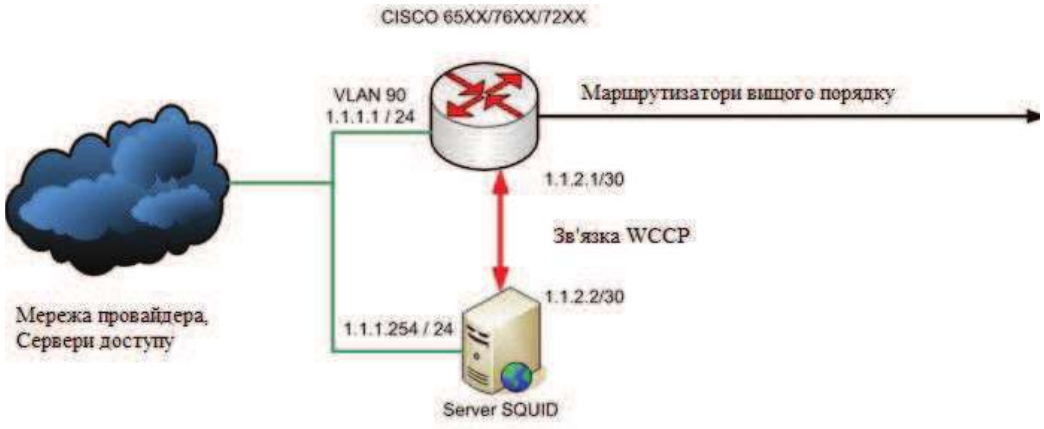

Рисунок 1 – Схема фільтрації мережевого трафіку

Трафік HTTP і HTTPS, що виходить з хмари мережі, що на рисунку 1 (в прикладі VLAN90) через встановлений бордер CISCO, будемо порівнювати з ACL, що містить IP-адреси, витягнуті зі списку. У разі збігу, трафік по протоколу WCCP (Web Cache Communication Protocol) відправляється на сервер (а) SQUID.

Дана схема дозволить уникнути аналізу всього трафіку і аналізувати тільки той, який йде на заявлені IP, що не може не позначитися позитивно на загальній продуктивності системи[3].

Одним з мінусів цієї схеми слід є те що, якщо припустимо внесений до реєстру ресурс з IP = 3.3.3.3, доменом = bad.xxx і URL = http://www.bad.xxx/1.jpg переїжджає на інший IP, то фільтр працювати не буде. Але з іншого боку актуалізація списків блокування, не відноситься до розгляду цієї задачі. Даний підхід дозволить з одного боку, не використовувкати дорогі системи аналізу трафіку, що проходить, з іншого

боку виключити «прямий» підхід блокування всього ресурсу по IP, а блокувати тільки зазначений в реєстрі контент.

Зв'язку з WCCP краще налаштовувати на приватних адресах, що б не писати зайві ACL в SQUID. Для початку потрібно отримати електронний підпис з реєстру сайтів, отриманого для фільтрації. Для цього скористаємося утилітою P12FromGostCSP, і експортуємо підпис в форматі PKSC # 12 [4].

Якщо експорт вдався, то можна перейти далі до побудови системи. якщо експорт не вдався, то автоматичне завантаження реєстру буде не доступне, і доведеться файл качати і викладати реєстр вручну.

Для системи фільтрації трафіку буде використовуватись сервер на базі FreeBSD

Необхідно знайти файл /usr/local/openssl/openssl.cnf і додати в кінець файлу:

```
[Openssl_def]
engines = engine_section
[Engine_section]
gost = gost_section
[Gost_section]
default_algorithms = ALL
engine_id = gost
#dynamic_path = /usr/local/lib/engines/libgost.so
CRYPT_PARAMS = id-Gost28147-89-CryptoPro-A-ParamSet
```

На початок цього файлу необхідно додати: openssl_conf = openssl_def

Виконаємо копіювання на сервер сертифікат в форматі PKSC # 12 і конвертуємо його в формат PEM: / Usr / local / bin / openssl pkcs12 -in p12.pfx -out cert.pem -nodes -clcerts

Готуємо структуру директорій для роботи скриптів і виконаємо копіювання туди сконвертований сертифікат:

```
mkdir / var / db / zapret-info /
mkdir / var / db / zapret-info / cfg
mkdir / var / db / zapret-info / arch
cp cert.pem / var / db / zapret-info / cfg /
mkdir / tftpboot
```

Необхідно створити файл запит відповідно до вимог (де дані між тегами змінені на дані вашої фірми):

```
echo '<operatorName> ТОВ "Роги і копита" </ <operatorName>'> /var/db/zapret-info/cfg/request.xml
echo '<inn> 00XXXXXXXXXX </ inn>' >> /var/db/zapret-info/cfg/request.xml
echo '<ogrn> XXXXXXXXXXXXX </ ogrn>' >> /var/db/zapret-info/cfg/request.xml
echo '<email> info@rogaikopita.ru </ email>' >> /var/db/zapret-info/cfg/request.xml
```

Залежно від розкладок системної консолі отриманий файл /var/db/zapret-info/cfg/request.xml необхідно конвертувати в кодування WINDOWS-1251[5].

Завдання даного скрипта - створити, підписати і відправити запит завантажити реєстр після відправки.

```
#! / usr / bin / perl -w
use strict;
use SOAP :: Lite;
use MIME :: Base64;
use Sys :: Syslog qw (: DEFAULT setlogsock);
use POSIX qw (strftime);
…
```

До завдань скрипта xml-paser.pl входить обробка отриманого реєстру та підготовка двох файлів - block.acl і denied.conf. Файл block.acl містить правила для cisco

і як правило завантажується по tftp. Файл denied.conf містить URL для редиректора SQUID. Скрипт аналізує, префікс http або https вказано в URL і у відповідності з даними префіксом генерує ACL для CISCO[6]. Для ресурсів з множинними IP і / або URL, скрипт генерує два ACL - як по порту www, так і по порту 443 (так як реєстр сам по собі сирої, і такі записи не передбачають однозначного трактування який протокол на якому IP використовувати).

На початку даного файлу треба встановити значення змінних, у відповідності з спеціальними налаштуваннями. Змінна $ tftp_root повинна бути встановлена в відповідно до настройками tftp-сервера, змінна $ query_ip повинна містити IP-адресу інтерфейсу сервера, через який надсилається в Інтернет фільтрований трафік[7], для запобігання утворенню кільцевої обробки запитів.

Для роботи скриптів нам потрібно встановити наступні порти:

cd / usr / ports / net / p5-SOAP-Lite; make install
cd / usr / ports / converters / p5-MIME-Base64; make install
cd / usr / ports / sysutils / p5-Sys-Syslog; make install
cd / usr / ports / textproc / p5-XML-Simple; make install
cd / usr / ports / devel / p5-Data-Dumper; make install
cd / usr / ports / converters / p5-Encode; make install
cd / usr / ports / net / p5-URI; make install
cd / usr / ports / converters / p5-Text-Iconv; make install

На даному етапі ми реалізували отримання і розбір реєстру сайтів, що необхідно фільтрувати. Надалі проведемо налаштування апартної частини сервера.

Виконаємо налаштуваня CISCO для роботи з сервером SQUID по WCCP. Група 0 використовується для відсилання http-трафіку, група 70 - для відсилання https-трафіку.

ip wccp 0 redirect-list WCCP_REDIRECT group-list 10 accelerated
ip wccp 70 redirect-list WCCP_REDIRECT group-list 10 accelerated
access-list 10 remark +++ WCCP_SQUID_PROXY +++
access-list 10 permit 1.1.2.2

Створюємо користувача, який буде вантажити access-листи і налаштовуємо rsh.

username blocker privilege 15 password password
no ip rcmd domain-lookup
ip rcmd rsh-enable
ip rcmd remote-host blocker 1.1.2.2 root enable

Ставимо з портів SQUID. При виборі опцій обов'язково контролюємо, що WCCP і WCCP2 протоколи включені. Створюємо конфіг (в прикладі для версії 3.2) /usr/local/etc/squid/squid.conf[8-9].

http_port 1.1.2.2:9090
http_port 1.1.2.2:3128 transparent
https_port 1.1.2.2:3129 transparent ssl-bump generate-host-certificates = on dynamic_cert_mem_cache_size = 4MB cert = / usr / local / etc / squid / squid.pem
always_direct allow all
ssl_bump allow all
…
acl QUERY urlpath_regex cgi-bin \\?
cache deny QUERY
acl apache rep_header Server ^ Apache
cache_mem 1 MB
…
dns_nameservers 8.8.8.8 8.8.4.4
hosts_file / etc / hosts
refresh_pattern ^ ftp 1440 20% 10080

```
refresh_pattern ^ gopher 1440 0% 1440
refresh_pattern. 0 20% 4320
quick_abort_min 0 KB
quick_abort_max 0 KB
half_closed_clients off
acl purge method PURGE
acl CONNECT method CONNECT
acl SSL_ports port 443
acl Safe_ports port 80
acl Safe_ports port 443 # https
acl Safe_ports port 1025-65535 # unregistered ports
http_access allow manager localhost
…
icp_access allow all
cache_mgr info@my.domain
cache_effective_group proxy
memory_pools off
log_icp_queries off
cachemgr_passwd q1w2e3r4 all
client_db off
buffered_logs on
wccp2_router 1.1.2.1
…
redirect_program /usr/local/etc/squid/redirector.pl
url_rewrite_children 20 startup = 10 idle = 1 concurrency = 0
```

Висновок. Було виконано базові налаштування сервера SQUID та маршрутизатора CISCO для фільтрації мережевого трафіку згідно запропонованої схеми. Розроблена схема дозволяє розмежовувати фільтрацію трафіку за контентом завдяки попередній фільтрації IP адрес за реєстром наданим для фільтрації ззовні. Запровадження каскадної фільтрації за IP адресами та за контентом дозволить зберегти високу пропускну здатність та високий рівень безпеки мережевого трафіку.

Перелік використаних джерел

1. Бикманс Герард. Linux from Scratch. Version 8.4, 2019. — 368 с.
2. Васильєва Н.К. та ін. Інформатика в LINUX-середовищі, Навч. посібник / кол. авт.; за ред. Н.К. Васильєвої. — Дніпропетровськ: Біла К., 2016. — 267 с.
3. Гончарук С.В. Администрирование ОС Linux, М.:НОИ Интуит , 2016. — 164 с.
4. Донцов В.П., Сафин И.В. Linux на примерах , СПб.: Наука и техника, 2017. — 352 с.
5. Керриск Майкл. Linux API. Исчерпывающее руководство, СПб.: Питер, 2018.- 1248с.
6. Колисниченко Д.Н. Linux. От новичка к профессионалу, 6-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2018. — 674 с.
7. Немет Эви, Снайдер Гарт, Хейн Трент, Уэйли Бен, Макин Дэн. Unix и Linux: руководство системного администратора, 5-е изд.: Пер. с англ. — СПб.: Диалектика, 2020. — 1168 с.
8. Пронин Я. Linux Inside на русском, Gitbook.com, 2018. — 847 с.
9. Стивенс Р., Раго С. UNIX. Профессиональное программирование, СПб.: Питер, 2018. — 944 с.