

Home-based Threat Intelligence

written by Mert SARICA | 1 October 2019

Those of you who read my articles will recall that in my post titled “Escape from Imprisonment“, I enthusiastically discussed the advantages of using a router packed with security features. As I mentioned in the article, I had started using the dnscrypt-proxy tool to encrypt DNS traffic (Dns over HTTPS – DoH).

In today’s world where thermostats are getting smarter, smart TVs are equipped with cameras, and electric water heaters and irons are being turned into spy devices, insecure Internet of Things (IoT) devices connected to our home network pose a great risk to our security and privacy. As I was thinking about how to detect systems in our home network that have been hacked, infected, or contain backdoors, I remembered that thanks to the dnscrypt-proxy tool, I could also record DNS requests made by all systems, devices, and gadgets connected to the home network.

At the point where I could record DNS requests, I realized I could detect malicious systems in my home network by querying the domain names and IP addresses found in these DNS requests through cyber threat intelligence services like Open Threat Exchange (OTX) and Critical Stack. Without wasting time, I started thinking about the list of requirements to bring this idea to life.

First, I decided to install the syslog-ng package on the Ubuntu operating system running on my Mini-PC, which is always at hand and always comes to my aid in such situations. After installing the package, I configured it to record incoming DNS requests in the date.log file under the /var/log/dns-sys/sender’s-ip-address directory and saved this configuration in the /etc/syslog-ng/conf.d/dns-sys.conf file.

```

root@ubuntu:/etc/syslog-ng/conf.d# ls
dns-sys.conf
root@ubuntu:/etc/syslog-ng/conf.d# cat dns-sys.conf
#####
options {
    create_dirs(yes);
    perm(0640);
    dir_perm(0750);
};

#####
source s_net {
    tcp(ip(0.0.0.0) port(514));
    udp(ip(0.0.0.0) port(514));
};

#####
destination d_host-specific {
    file("/var/log/dns-sys/$HOST/$DAY-$MONTH-$YEAR.log");
};

filter f_cached { match("cached"); }; # Filter regex keyword cached
filter f_query { match("query"); }; # Filter regex keyword query
filter f_reply { match("reply"); }; # Filter regex keyword reply

log {
    source(s_net);
    filter(f_cached);
    destination(d_host-specific);
};

log {
    source(s_net);
    filter(f_query);
    destination(d_host-specific);
};

log {
    source(s_net);
    filter(f_reply);
    destination(d_host-specific);
};

```

In the next step, to make the dnscrypt-proxy tool log DNS requests to the router's syslog, I added the line 'log-queries' to the /jffs/configs/dnsmasq.conf.add file. Then, to make the router display these requests on its syslog page, I set the 'Default message log level' and 'Log only messages more urgent than' values to 'debug', and to redirect these messages to the syslog-ng application running on Ubuntu, I defined the 'Remote Log Server' value as the IP address of Ubuntu.

```

mert@RT-AC1900U-6610:/jffs/configs# cat dnsmasq.conf.add
no-resolv
log-queries
server=127.0.0.1#65053
mert@RT-AC1900U-6610:/jffs/configs# █

```

ASUS RT-AC1900U Powered by Asuswrt-Merlin Logout Reboot English

Operation Mode: **Wireless router** Firmware Version: **384.9** SSID: [REDACTED]

General Log Wireless Log DHCP leases IPv6 Routing Table Port Forwarding Connections

System Log - General Log

This page shows the detailed system's activities.

System Time	Wed, Mar 27 21:07:46 2019
Uptime	17 days 9 hours 34 minute(s) 20 seconds
Remote Log Server	192.168.1. Port: 514
Default message log level	debug
Log only messages more urgent than	debug

Apply

Auto refresh

```

Mar 27 21:07:40 dnsmasq[29860]: reply wildcard-ru.asustek.com.akadns.net is <CNAME>
Mar 27 21:07:40 dnsmasq[29860]: reply e11960.dace15.akamaiedge.net is 104.101.244.165
Mar 27 21:07:40 dnsmasq[29860]: dnssec-query[DS] trafficmanager.net to 127.0.0.1
Mar 27 21:07:40 dnsmasq[22450]: dnssec-query[DNSKEY] ca to 127.0.0.1
Mar 27 21:07:40 dnsmasq[22450]: reply ca is DNSKEY keytag 48662, algo 8
Mar 27 21:07:40 dnsmasq[22450]: reply ca is DNSKEY keytag 2134, algo 8
Mar 27 21:07:40 dnsmasq[22450]: reply ca is DNSKEY keytag 43854, algo 8
Mar 27 21:07:40 dnsmasq[22450]: reply ca is DNSKEY keytag 35433, algo 8
Mar 27 21:07:40 dnsmasq[22450]: reply lostrealm.ca is DS keytag 2371, algo 13, digest 2
Mar 27 21:07:40 dnsmasq[22450]: validation result is SECURE
Mar 27 21:07:40 dnsmasq[22450]: reply asuswrt.lostrealm.ca is 174.142.221.134
Mar 27 21:07:40 dnsmasq[29860]: reply trafficmanager.net is no DS
Mar 27 21:07:40 dnsmasq[29860]: validation result is INSECURE
Mar 27 21:07:40 dnsmasq[29860]: reply account.asus.com is <CNAME>
Mar 27 21:07:40 dnsmasq[29860]: reply asusaccount.trafficmanager.net is <CNAME>
Mar 27 21:07:40 dnsmasq[29860]: reply ssoap.japanwest.cloudapp.azure.com is 138.91.27.92
Mar 27 21:07:44 dnsmasq[29860]: query[AAAA] google.com from 127.0.0.1
Mar 27 21:07:44 dnsmasq[29860]: cached google.com is 2607:f8b0:4002:811::200e
Mar 27 21:07:44 dnsmasq[29860]: query[A] google.com from 127.0.0.1
Mar 27 21:07:44 dnsmasq[29860]: cached google.com is 172.217.0.78
Mar 27 21:07:44 dnsmasq[29860]: query[PTR] e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.1.8.0.2.0.0.4.0.b.8.f.7.0.6.2.ip6
Mar 27 21:07:44 dnsmasq[29860]: cached 2607:f8b0:4002:811::200e is at126s14-in-x0e.1e100.net
Mar 27 21:07:44 dnsmasq[29860]: query[PTR] 78.0.217.172.in-addr.arpa from 127.0.0.1
Mar 27 21:07:44 dnsmasq[29860]: cached 172.217.0.78 is at126s16-in-f14.1e100.net
Mar 27 21:07:44 dnsmasq[29860]: cached 172.217.0.78 is nuq04s19-in-f14.1e100.net

```

Clear Save

I started examining the syslog-ng records one by one and looking into which types of records I needed to focus on for threat intelligence. After learning that I could use the query[A], cached, and reply information in the records, I thought I could send these records to Security Onion, which integrates with OTX. After installing and running Security Onion's 16.04.5.6 operating system, I noticed that the logstash service (so-logstash) wasn't working at all. Despite my struggle, I was unsuccessful and started researching alternative methods.

```

root@ubuntu:/etc/syslog-ng/conf.d# tail -n 20 /var/log/dns-sys/192.168.1.1/09-04-2019.log
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply partnerad.l.doubleclick.net is 74.125.21.156
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply partnerad.l.doubleclick.net is 74.125.21.157
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply partnerad.l.doubleclick.net is 74.125.21.154
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply partnerad.l.doubleclick.net is 74.125.21.155
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: query[A] s.w.org from 192.168.1.225
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: query[A] widget.engageya.com from 192.168.1.225
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply s.w.org is 192.0.77.48
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply widget.engageya.com is <CNAME>
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply widget-engageya.edgekey.net is <CNAME>
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply e15247.dscg.akamaiedge.net is 104.96.141.105
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: query[A] www.googletagservices.com from 192.168.1.225
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply www.googletagservices.com is <CNAME>
Apr 9 21:09:25 192.168.1.1 dnsmasq[29860]: reply pagead46.l.doubleclick.net is 172.217.3.226
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: query[A] gatr.hit.gemius.pl from 192.168.1.225
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 5.135.121.144
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 37.59.195.0
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 37.187.168.211
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 149.202.193.219
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 149.202.204.241
Apr 9 21:09:27 192.168.1.1 dnsmasq[29860]: reply gatr.hit.gemius.pl is 188.165.145.88
root@ubuntu:/etc/syslog-ng/conf.d# cat /var/log/dns-sys/192.168.1.1/08-04-2019.log | cut -d " " -f 7 | sort | uniq -i
cached
dnssec-query[DNSKEY]
dnssec-query[DS]
forwarded
query[A]
query[AAAA]
query[PTR]
query[SRV]
reply
root@ubuntu:/etc/syslog-ng/conf.d#

```

When I shared a message on Twitter about needing to install ELK, I received messages suggesting that I could use cloud and ready-made ELK systems. As I was considering whether to install ELK on Ubuntu or use a cloud system, I learned that Logstash, which has Grok filter and Translate filter plugins, was tailor-made for this job.



Mert SARICA @MertSARICA · 7 Mar

Yapılacaklar listem kabardıkça kabanyor, eve gidince ELK kurmam lazım. Beni bu kadar çok çalıştıran kendimi, şikayet edecek bir merci bulmam lazım. :)

3



11



Furkan ÇALIŞKAN

@caliskanfurkan_

Takip ediliyor

@MertSARICA adlı kullanıcıya yanıt olarak

cloud.elastic.co 14 gün ücretsiz hazır cloud ELK :)

22:47 - 7 Mar 2019

5 Beğeni



1



5



Yanıtını Tweetle



Mert SARICA @MertSARICA · 7 Mar

@caliskanfurkan_ adlı kullanıcıya yanıt olarak

Eyv.

1



Samet @belleveben · 8 Mar

Bu da docker elk. elk-docker.readthedocs.io

2



2



I started modifying the securityonion-otx script file, which was developed for Security Onion – OTX integration, according to my needs. I set the bro-otx file to save threat intelligence information from OTX to the /etc/logstash/ls-otx/otx.dat file every hour. I also configured the OTX.py file to extract only domain name information from the malicious URL and DOMAIN entries in the otx.dat file and save it as the

/etc/logstash/translate/OTX.yaml file to be read by the Translate filter at the 5th minute of every hour.

```
root@ubuntu:/etc/cron.d# cat bro-otx
# /etc/cron.d/bro-otx
#
# crontab entry to manage Bro OTX pulse updates

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

0 * * * * root python /etc/logstash/ls-otx/bro-otx.py >> /var/log/bro-otx.log 2>&1
root@ubuntu:/etc/cron.d# cat ls-otx
# /etc/cron.d/bro-otx
#
# crontab entry to create Logstash dictionary from OTX file

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

*/5 */1 * * * * root python /etc/logstash/ls-otx/OTX.py >> /var/log/ls-otx.log 2>&1
root@ubuntu:/etc/cron.d#
```

```
GNU nano 2.9.3 otx.dat
#Fields indicator indicator_type meta_source meta.url meta.do_notice
34ba8798c01b452d708c1409590ea30 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
4601e75267d0dcf7e2456c3f45ec470a Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
76c173d469c3a7313a5c032314256c Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
803bf506e55ab736f4c018d15739e352 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
F547e6f4376d0873f2f02b91e40230 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
827f8e92b43d13c0a3368c37735d178b6e85d136231e69fe02df Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
www.uscni1ers.com/r/m/images/01/js/index.php Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
www.lumw.com/wp-includes/images/site.php Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
db909c50b4f263ef7690289680a37f Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
c0ec10a8b0525ba10254b87f406e36 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
224652a8a0683213e6f51457f7e020 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
50e86b6c5cf9a4bf97345935725f20f Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
6b5ce7fb6dd1e588f8d1c344720f7c7a Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
c7323e35841980e38129b3a5a000da Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
73c79f84361fc8d74ec53c36e07b396e Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
72464752864933dc64b03e46d84c9f0 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
181d4f01c6d01aba0847cc74e24268 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
95a2287f560b1b9f98a131a13558b Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
patane.myon1neportal.org Intel: DOMAIN AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_Threat_Group.pds
isozaki.sakura.ne.jp/p1c1c Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_Threat_Group.pds
patane.myon1neportal.org Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_Threat_Group.pds
www.wco-kyousoai.com/ex-engine/modules/comment/queries/deletecomment.php Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
www.sics.net.zy/images/pat/terms/preview/deletecomments.php Intel: URL AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
51c11bf10ee6e631d1970863c41a1393 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
57d126b7ad05af9d9e0e9946bb9e65 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
f92f8bd9442cd2eb3a3e688cc75 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
a287d487eed8f4ce4ba1ca54708f3 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
ec0ef96943300ef5030245b20bc706 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
67b27d0fb06e08a60f0c16b93b0e7 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
59423b229724c2e7294b01a2f81c21 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
15898d0671637094007395d6f42238b Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
e4d2342304341ea20ed01c028404d Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
e4f61f03d8ced007f38644893883c Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
0e9def2304dae5f95dc1c50774f889b37 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
0fdd2b0cf506661cf4d05f9fcb9e61 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
50de060f1689856311eb97c5c1d0d3 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
2520c206294045c9212be8e25211599 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
e98113a8957190a0c12c714f00689 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
1f4904dac4f15d97293c86393203f Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
8090282a98f035b0778de6884d7720c0 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
753ac3700a31f8a68f9e8493850f72d8 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
1f2a249430583b89cf72d0745370 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
8e60d4502c34610b833e33f91c5728 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
e43996f45cb889a30e443732973a22 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
3dce29291a344b4ef9729404f527704 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
054cfff8c56245c54793379f17b19 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
08d65177026f49e55d011d8a117c4c8 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
c4c068126a11c1e608f3886ad5f779 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
61654e2ea3c22beaf44ef50b71f57 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
d0ef0b3f10f4723eeac332ad7f38f Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
db1bc042be04ae1add09ab50bd1c9d Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
893f4b3c99c3805db0e61e1c9e7980d Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
da0683bb5e66180313618e772d5f5 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
1c2bbe6e33f01e81be5998d08a38b Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
b1080f0bd16868f27b00deaf73733e Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
e7106819a131419633054c03e390d Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
418b22173deb86d4a958d1418f8fd Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
47f507466e95c24e7002529f025 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
a4382cf7110be0183a34c91369f81 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
a92f17f5cccf378a6a4e8f239acd9 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
068aae09a272244a5b9f0d5430109 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
3c6e67f0c068183637d4ade90757a84 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
82cef22a4f4abb7e05c078e9dedc93 Intel: FILE_HASH AlienVault OTX2 - Tick Group ID: 5cald06890ff2a34699adb68 Author: AlienVault http://download.ahnlab.com/kr/site/library/Analysis_Report/Tick_S
```



```

root@ubuntu:/etc/logstash/ls-otx# cat OTX.py
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# OTX to Logstash Dictionary Script
# Author: Mert SARICA
# E-mail: mert [ . ] sarica [ @ ] gmail [ . ] com
# URL: https://www.mertsarica.com
#
# Credit: https://raw.githubusercontent.com/TravisFSmith/MyBroElk/master/maliciousIP.py

import re
debug = 0

def writeYAML():
    fname = "/etc/logstash/ls-otx/otx.dat"
    yamlFile = open('/etc/logstash/translate/OTX.yaml', 'w')
    with open(fname) as html:
        cti = []
        for line in html.readlines():
            line = re.sub('\r|\n', '', line)
            if line.find("Intel::DOMAIN") >= 0:
                try:
                    line = line.split("\t")[0]
                    if line not in cti:
                        cti.append(line)
                        if debug:
                            print line.split("\t")[0]
                    yamlFile.write("\t" + line + "\t": \"YES\" + "\n")
                except:
                    continue

            if line.find("Intel::URL") >= 0:
                try:
                    line = line.split("\t")[0]
                    line = line.split("/")[0]
                except:
                    line = line.split("\t")[0]

                try:
                    line = line.split(":")[0]
                    if line not in cti:
                        cti.append(line)
                        if debug:
                            print line
                    yamlFile.write("\t" + line + "\t": \"YES\" + "\n")
                except:
                    if line not in cti:
                        cti.append(line)
                        if debug:
                            print line
                    yamlFile.write("\t" + line + "\t": \"YES\" + "\n")

    yamlFile.close()

if __name__=="__main__":
    writeYAML()
root@ubuntu:/etc/logstash/ls-otx# █

```

```

root@ubuntu:/etc/logstash/translate# ls
OTX.yaml
root@ubuntu:/etc/logstash/translate# head -n 10 OTX.yaml
"www.aucsellors.com": "YES"
"www.lunwe.com": "YES"
"patane.myonlineportal.org": "YES"
"isozaki.sakura.ne.jp": "YES"
"www.wco-kyousai.com": "YES"
"www.51cs.net": "YES"
"www6.intarnetservice.com": "YES"
"www.webmailerservices.com": "YES"
"go-trust.webmailerservices.com": "YES"
"www.adobeservice.net": "YES"
root@ubuntu:/etc/logstash/translate#

```

In the configuration file of Logstash (logstash.conf), I defined the rules to read DNS records logged by syslog-ng with the Grok filter and to send an alert via email if any of the IP addresses or domain names in these records match with those in the OTX.yaml file using the Translate filter. Then I restarted Logstash and made an nslookup for the address www[.]aucsellors[.]com listed in the OTX.yaml file. With this, the alert was

successfully generated and sent to me by email, and I had successfully implemented the home-based threat intelligence service. :)

Some log lines you want to match. It's helps much to use several lines, and to choose lines that are as diverse as possible.

```
Mar 27 20:15:31 192.168.1.1 dnsmasq[29860]: reply upu.samsungelectronics.com is 54.83.144.140
```

The (unquoted) pattern that should match all logfile lines.(Please keep in mind that the whole log line / message is searched for this pattern; if you want this to match the whole line, enclose it in ^ s or ^ A Z. This speeds up the search - especially if the pattern is not found.)

```
%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}?(?:[%POSINT:syslog_pid])? (reply|cached) % (GREEDYDATA:syslog_iporhost) (is) %{GREEDYDATA:syslog_iporhost}
```

Please mark the libraries of grok Patterns from logstash v2.4.0 which you want to use. You probably want to use grok-patterns if you use any of the others, since they rely on the basic patterns defined there.

- firewalls
- aws
- bro
- exim
- bind
- haproxy
- linux-syslog
- squid
- mcollective-patterns
- bacula
- postgresql
- java
- maven
- grok-patterns
- htpfd
- redis
- nagios
- rails
- mongodb
- ruby
- mcollective
- junos

You can also provide a library of some additional grok patterns in the same format as the pattern files linked above. On each line you give a pattern name, a space and the pattern. For example: WORD %w-%b

If you want to use logstash's multiline filter please specify the used pattern (can include grok Patterns):

negate the multiline regex

Mar 27 20:15:31 192.168.1.1 dnsmasq[29860]: reply upu.samsungelectronics.com is 54.83.144.140	
malicious	
syslog_program	dnsmasq[29860]
syslog_hostname	192.168.1.1
syslog_iporhost2	54.83.144.140
syslog_iporhost	upu.samsungelectronics.com
syslog_timestamp	Mar-27-20:15:31

```
root@ubuntu:/etc/logstash# cat logstash.conf
input {
  # stdin { type => syslog }
  file {
    path => "/var/log/dns-sys/192.168.1.1/*.log"
    start_position => "beginning"
  }
}

filter {
  grok {
    match => { "message" => "%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}(?:\[%(POSINT:syslog_pid)\])?: (reply|cached) %{GREEDYDATA:syslog_iporhost} (is) %{GREEDYDATA:syslog_iporhost2}" }
    add_tag => "dnsmasq"
  }
  grok {
    match => { "message" => "%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}(?:\[%(POSINT:syslog_pid)\])?: (query\[A\]) %{GREEDYDATA:syslog_iporhost} (from) %{GREEDYDATA:syslog_queryfrom}" }
    add_tag => "dnsmasq"
  }
  translate {
    field => "syslog_iporhost"
    destination => "malicious"
    dictionary_path => "/etc/logstash/translate/otx.yaml"
    add_tag => "malicious"
  }
  translate {
    field => "syslog_iporhost2"
    destination => "malicious"
    dictionary_path => "/etc/logstash/translate/otx.yaml"
    add_tag => "malicious"
  }
  mutate {
    remove_tag => ["_grokparsefailure"]
  }
  if "dnsmasq" not in [tags] {
    drop { }
  }
}

output {
  stdout {
    codec => rubydebug
  }
  if [malicious] == "YES" and [syslog_iporhost2] {
    email {
      address => "127.0.0.1"
      from => "alert@mertsarica.com"
      htmlbody => "Malicious traffic has been detected!<br/><br/>
      <b>Destination Domain: </b>{syslog_iporhost}<br/>
      <b>Destination IP: </b>{syslog_iporhost2}<br/>
      <b>Raw Log: </b>{message}"
      port => 25
      subject => "Malicious Traffic"
      to => "mert_sarica@gmail.com"
      use_tls => false
    }
  }
  else if [malicious] == "YES" and [syslog_queryfrom] {
    email {
      address => "127.0.0.1"
      from => "alert@mertsarica.com"
      htmlbody => "Malicious traffic has been detected!<br/><br/>
      <b>Source IP: </b>{syslog_queryfrom}<br/>
      <b>Destination IP or Domain: </b>{syslog_iporhost}<br/>
      <b>Raw Log: </b>{message}"
      port => 25
      subject => "Malicious Traffic"
      to => "mert_sarica@gmail.com"
      use_tls => false
    }
  }
}
```



```
root@ubuntu:/etc/logstash# /usr/share/logstash/bin/logstash -f /etc/logstash.conf
WARNING: could not find logstash.yml which is typically located in $LS_HOME/config or /etc/logstash. You can specify the path using --path.settings. Continuing using the defaults
Could not find log4j2 configuration at path /usr/share/logstash/config/log4j2.properties. Using default config which logs errors to the console
[WARN ] 2019-04-01 21:47:48.681 [Logstash:runner] multi/local - ignoring the 'pipelines.yml' file because modules or command line options are specified
[INFO ] 2019-04-01 21:47:48.747 [Logstash:runner] runner - starting Logstash {"logstash.version"=>"6.7.0"}
[INFO ] 2019-04-01 21:48:36.336 [Converge PipelineAction::create<main>] pipeline - Starting pipeline {:pipeline_id=>"main", :pipeline_workers=>4, :pipeline_batch_size=>125, :pipeline_batch_delay=>50}
[INFO ] 2019-04-01 21:48:46.924 [Converge PipelineAction::create<main>] pipeline - Pipeline started successfully {:pipeline_id=>"main", :thread=>#<Thread:0x4aee8f3b run>}
The stdin plugin is now waiting for input.
[INFO ] 2019-04-01 21:48:47.157 [Ruby-0-Thread-1: /usr/share/logstash/lib/bootstrap/environment.rb:6] agent - Pipelines running {:count=>1, :running_pipelines=>[:main], :non_running_pipelines=>[]}
[INFO ] 2019-04-01 21:48:48.417 [api webserver] agent - Successfully started Logstash API endpoint {:port=>9600}
Mar 31 19:17:49 192.168.1.1 dnsmasq[29860]: reply test.com is 173.194.219.138
/usr/share/logstash/vendor/bundle/ruby/2.3.0/gems/awesome_print-1.7.0/lib/awesome_print/formatters/base_formatter.rb:31: warning: constant ::Fixnum is deprecated
{
  "syslog_program" => "dnsmasq",
  "message" => "Mar 31 19:17:49 192.168.1.1 dnsmasq[29860]: reply test.com is 173.194.219.138",
  "host" => "0.0.0.0",
  "syslog_iporhost" => "test.com",
  "syslog_pid" => "29860",
  "token" => "omTayqWxwxyroesSsitgNZLYnXkva",
  "timestamp" => "2019-04-01T18:49:17.281Z",
  "syslog_iporhost2" => "173.194.219.138",
  "type" => "syslog",
  "syslog_hostname" => "192.168.1.1",
  "syslog_timestamp" => "Mar 31 19:17:49",
  "tags" => [
    [0] "dnsmasq"
  ],
  "version" => "1"
}
Mar 31 19:17:49 192.168.1.1 dnsmasq[29860]: reply www.aucselllers.com is 173.194.219.138
{
  "syslog_program" => "dnsmasq",
  "message" => "Mar 31 19:17:49 192.168.1.1 dnsmasq[29860]: reply www.aucselllers.com is 173.194.219.138",
  "host" => "0.0.0.0",
  "syslog_iporhost" => "www.aucselllers.com",
  "syslog_pid" => "29860",
  "token" => "omTayqWxwxyroesSsitgNZLYnXkva",
  "timestamp" => "2019-04-01T18:49:27.866Z",
  "syslog_iporhost2" => "173.194.219.138",
  "type" => "syslog",
  "syslog_hostname" => "192.168.1.1",
  "syslog_timestamp" => "Mar 31 19:17:49",
  "tags" => [
    [0] "dnsmasq",
    [1] "malicious"
  ],
  "version" => "1",
  "malicious" => "YES"
}
```



Malicious Traffic Inbox x



alert@mertsarica.com via [sandbox.mgsend.net](#)

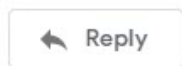
to me ▾

Malicious traffic has been detected!

Destination Domain: www.aucselllers.com

Destination IP: 173.194.219.138

Raw Log: Mar 31 19:17:49 192.168.1.1 dnsmasq[29860]: reply www.aucselllers.com is 173.194.219.138



Hope to see you in the following articles.