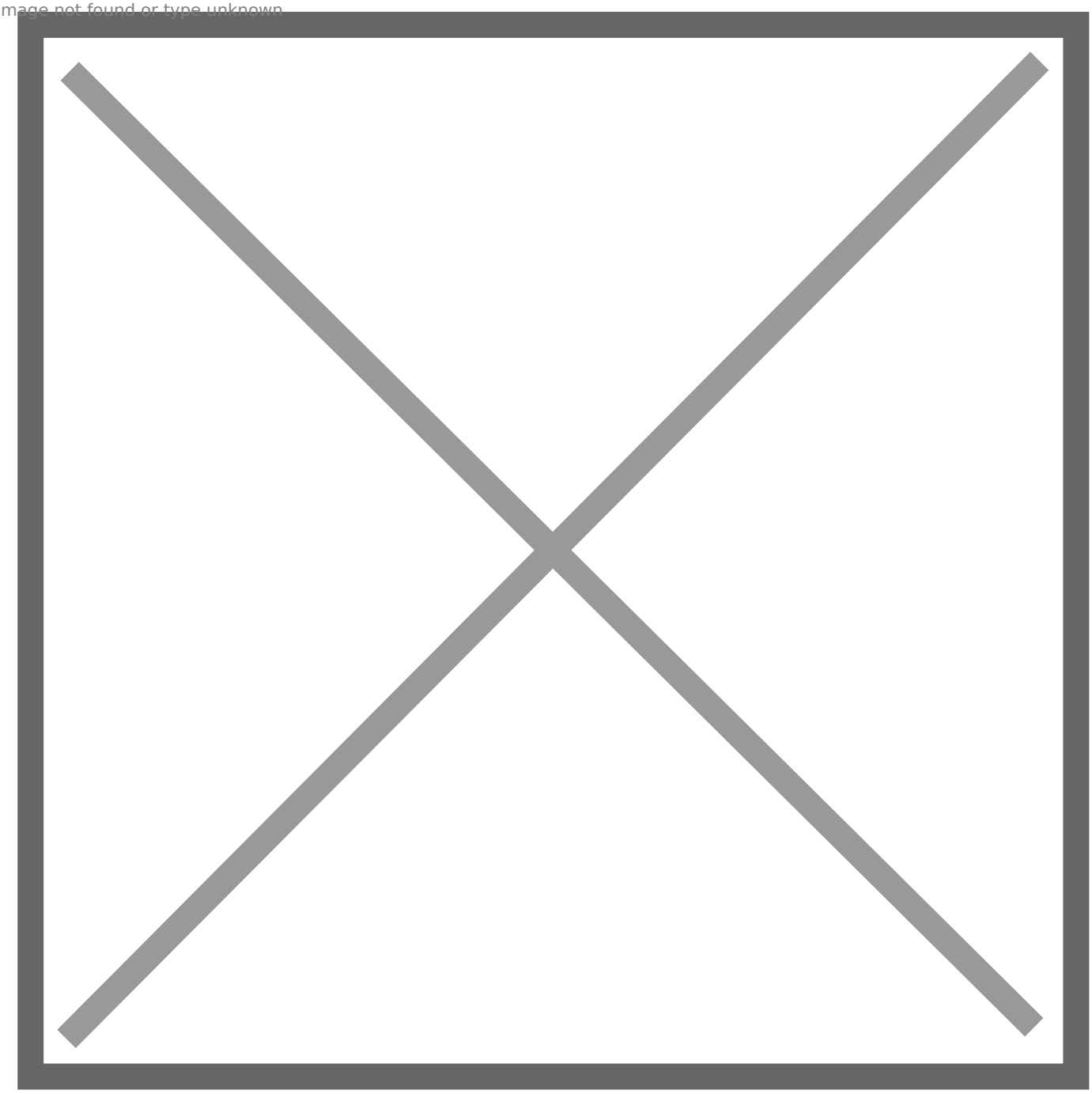
Как установить Suricata на FreeBSD

7 февраля 2020 года от Альберта Вальбуэны

Image not found or type unknown



Suricata — это бесплатная система обнаружения вторжений с открытым исходным кодом, или сокращённо IDS. Но она также может выступать в качестве системы предотвращения вторжений, или IPS. Она работает на основе эвристических алгоритмов, выявляющих закономерности в сетевом трафике. Если система настроена только на предупреждение о подозрительной активности, она называется IDS, а если она блокирует трафик из-за вредоносной активности, то IPS. Suricata обычно устанавливается в качестве плагина в pfSense — полноценный брандмауэр и сетевой дистрибутив корпоративного уровня с открытым исходным кодом на базе FreeBSD. Если вы используете FreeBSD в качестве настольной системы, вот руководство по тестированию pfSense в VirtualBox. Однако вы можете использовать Suricata как отдельное программное обеспечение для анализа сетевого трафика.

Что касается требований к установке Suricata, то их немного. Вы можете использовать ее на том же сервере, на котором работает веб-сервер, но гораздо интереснее использовать ее вместе с брандмауэром, защищающим офис (локальная сеть) или даже компанию с несколькими филиалами (глобальная сеть). Если вы решите использовать ее для защиты глобальной сети, то вам практически обязательно понадобится сервер или устройство с двумя сетевыми интерфейсами. Конечно, если вы планируете защищать компанию среднего или крупного размера, то чем больше сетевых адаптеров, тем лучше, хотя вы также можете рассмотреть вариант с оборудованием Netgate, на котором уже установлен pfSense. Что бы вы ни использовали, обязательным условием является зеркальное отображение трафика на устройстве, на котором вы собираетесь использовать Suricata, если вы запускаете ее параллельно с другими устройствами. Обычно это означает, что нужно подключить кабель от коммутатора или брандмауэра к устройству Suricata и настроить зеркалирование портов. Если вы хотите использовать его на том же веб-сервере, что и раньше, то можете это сделать, но, скорее всего, вам больше подойдет HIDS (система обнаружения вторжений на уровне хоста).

Но зачем все это? Возможно, вам не нужна эта система обнаружения вторжений, но если у вас небольшой офис, то установка межсетевого экрана pfSense с пакетом Suricata (напомним, что pfSense полностью основан на FreeBSD) станет дополнительным уровнем безопасности. Опять же, для веб-сервера это не очень полезно, если только это не очень популярный веб-сервер и от него не зависит компания из 15 человек. В таком случае это программное обеспечение поможет защитить сеть, для чего оно и предназначено. Вопрос в том, как отображаются оповещения Suricata. Такое программное обеспечение обычно используется в средних и крупных организациях (опять же, в небольших компаниях можно использовать pfSense или нанять кого-то, кто сделает всю работу за вас). Обычно оповещения записываются в системный журнал и пересылаются в SIEM, специализированное программное обеспечение для анализа инцидентов. SIEM с открытым исходным кодом — это OSSIM. Но поверх Suricata можно установить стек ELK, который добавит графические возможности и позволит легко работать с оповещениями без необходимости установки SIEM. Об этом мы расскажем в одной из статей в этом году.

Для демонстрации мы используем виртуальную машину DigitalOcean, но вы можете использовать любой VPS по своему выбору или просто физическое оборудование. Учитывайте размер устройства или виртуальной машины, которые вы планируете использовать в производственной среде, поскольку Suricata может потреблять довольно много ресурсов, особенно в загруженных сетях.

Сначала мы поищем пакет. Если у вас установлены последние версии пакетов из репозитория FreeBSD, то, скорее всего, вы найдете только Suricata 5. Однако если вы используете ежеквартальные обновления, то найдете Suricata версий 4.1.6 и 5. Выбирайте на свой вкус, но я обычно устанавливаю последнюю версию.

```
[albert@VPN ~]$ pkg search suricata

suricata-5.0.1 High Performance Network IDS, IPS and Security Monitoring engine

suricata5-5.0.0.r1_2 High Performance Network IDS, IPS and Security Monitoring engine(v5)

[albert@VPN ~]$
```

Найдя пакет, мы просто установим его. Если сомневаетесь, посетите Freshports.org, где можно найти всю необходимую информацию о программном обеспечении для FreeBSD.

```
[albert@VPN ~]$ sudo pkg install suricata

Contrasenya:

Updating FreeBSD repository catalogue...

FreeBSD repository is up to date.

All repositories are up to date.

The following 8 package(s) will be affected (of 0 checked):
```

```
New packages to be INSTALLED:

suricata: 5.0.1

libyaml: 0.2.2

libnet: 1.1.6_5,1

python37: 3.7.6

py37-yaml: 5.2

py37-setuptools: 41.4.0_1

pcre: 8.43_2

jansson: 2.12

Number of packages to be installed: 8

The process will require 128 MiB more space.

20 MiB to be downloaded.

Proceed with this action? [y/N]: y

.....

.....

You may want to try BPF in zerocopy mode to test performance improvements:

sysctl -w net.bpf.zerocopy_enable=1

Don't forget to add net.bpf.zerocopy_enable=1 to /etc/sysctl.conf

[albert@VPN ~]$
```

Как всегда, полезно сохранить сообщения об установке, потому что они могут оказаться очень полезными.

```
=====

Message from suricata-5.0.1:

--

If you want to run Suricata in IDS mode, add to /etc/rc.conf:

suricata_enable="YES"

suricata_interface="<if>"
```

```
NOTE: Declaring suricata_interface is MANDATORY for Suricata in IDS Mode.

However, if you want to run Suricata in Inline IPS Mode in divert(4) mode,

add to /etc/rc.conf:

suricata_enable="YES"

suricata_divertport="8000"

NOTE:

Suricata won't start in IDS mode without an interface configured.

Therefore if you omit suricata_interface from rc.conf, FreeBSD's

rc.d/suricata will automatically try to start Suricata in IPS Mode

(on divert port 8000, by default).

Alternatively, if you want to run Suricata in Inline IPS Mode in high-speed

netmap(4) mode, add to /etc/rc.conf:

suricata_enable="YES"

suricata_netmap="YES"

NOTE:

Suricata requires additional interface settings in the configuration

file to run in netmap(4) mode.

RULES: Suricata IDS/IPS Engine comes without rules by default. You should

add rules by yourself and set an updating strategy. To do so, please visit:

http://www.openinfosecfoundation.org/documentation/rules.html

http://www.openinfosecfoundation.org/documentation/emerging-threats.html

You may want to try BPF in zerocopy mode to test performance improvements:

sysctl -w net.bpf.zerocopy_enable=1

Don't forget to add net.bpf.zerocopy_enable=1 to /etc/sysctl.conf
```

Для начала выясним, какой у нас интерфейс, с помощью команды ifconfig. Поскольку это виртуальная машина FreeBSD DOcean по умолчанию, у нее только один интерфейс (не считая loop-интерфейса). Рекомендуется использовать один интерфейс для управления, а второй — для приема зеркалированного трафика с коммутатора или брандмауэра. Если по

какой-то причине зеркалированный трафик превысит пропускную способность системы, у вас всегда будет запасной вариант. Для демонстрации мы будем использовать только один интерфейс, но мы вас предупредили. И вы это знаете.

```
[albert@VPN ~]$ ifconfig

vtnet0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500

options=6c07bb<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU,VLAN_HWCSUM,TSO4,TSO6,LRO,VLAN_HWTSO,LINKSTATE,RXCSUM_IPV6,TXCSUM_IPV6>

ether 2e:d3:db:28:6a:d8

inet6 fe80::2cd3:dbff:fe28:6ad8%vtnet0 prefixlen 64 scopeid 0x1

inet 142.93.75.244 netmask 0xfffff000 broadcast 142.93.79.255

inet 10.17.0.5 netmask 0xffff0000 broadcast 10.17.255.255

media: Ethernet 10Gbase-T <full-duplex>

status: active

nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>

lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384

options=680003<RXCSUM,TXCSUM,LINKSTATE,RXCSUM_IPV6,TXCSUM_IPV6>

inet6 ::1 prefixlen 128

inet6 fe80::1%lo0 prefixlen 64 scopeid 0x2

inet 127.0.0.1 netmask 0xff000000

groups: lo

nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>

[albert@VPN ~]$
```

Как мы видим, сетевой интерфейс здесь называется vtnet0. Давайте настроим этот интерфейс в режиме promiscuous. Зачем? Потому что мы хотим проверять сетевой трафик и хотим, чтобы каждый сетевой пакет проверялся полностью, а не только фреймы.

```
[albert@VPN ~]$ sudo ifconfig vtnet0 promisc

[albert@VPN ~]$
```

Чтобы проверить, что изменения вступили в силу, выполните следующую команду и найдите строку ‘promisc’.

```
[albert@VPN ~]$ ifconfig vtnet0 | grep 'PROMISC'

vtnet0: flags=28943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST,PPROMISC> metric 0 mtu 1500

[albert@VPN ~]$
```

Вот и все, теперь интерфейс vtnet0 передает все полученные пакеты в ядро, где они будут проверены.

Теперь мы можем включить Suricata как службу и настроить ее для работы на интерфейсе vtnet0. Мы сделаем это с помощью команды sysrc.

```
[albert@VPN ~]$ sudo sysrc suricata_enable="YES"

suricata_enable: -> YES

[albert@VPN ~]$
```

Теперь, когда служба добавлена, давайте настроим ее для работы на интерфейсе vtnet0.

```
[albert@VPN ~]$ sudo sysrc suricata_interface="vtnet0"

suricata_interface: -> vtnet0

[albert@VPN ~]$
```

Чтобы убедиться, что изменения внесены в файл /etc/rc.conf, просто введите следующую команду. Должны появиться две записи:

```
[albert@VPN ~]$ grep -n 'suricata' /etc/rc.conf

46:suricata_enable="YES"

47:suricata_interface="vtnet0"

[albert@VPN ~]$
```

Как видно выше, в строках 46 и 47 файла /etc/rc.conf указаны служба и интерфейс suricata.

Прежде чем мы запустим сервис и пакеты, проходящие через интерфейс, начнут проверяться, нужно настроить несколько параметров, таких как адрес электронной почты, на который будут приходить оповещения, расположение правил и ответы на некоторые другие вопросы о работе Suricata.

Suricata использует правила для проверки и работы в качестве системы обнаружения вторжений. Поэтому, если вы используете ее для защиты локальной сети, убедитесь, что в брандмауэре настроено правило для получения необходимых источников, иначе Suricata не сможет получать информацию и выполнять свою основную функцию. Правила находятся в следующем каталоге и файле:

```
/var/lib/suricata
```

```
/var/lib/suricata/suricata.rules
```

Правила также необходимо сортировать по приоритетам для уровней оповещений. Эта информация хранится в файле /usr/local/etc/suricata/classification.config. Следующий абзац взят из этого файла.

```
# $Id$
```

```
# classification.config taken from Snort 2.8.5.3. Snort is governed by the GPLv2
```

```
#
```

```
# The following includes information for prioritizing rules
```

```
#
```

```
# Each classification includes a shortname, a description, and a default
```

```
# priority for that classification.
```

```
#
```

```
# This allows alerts to be classified and prioritized. You can specify
```

```
# what priority each classification has. Any rule can override the default
```

```
# priority for that rule.
```

```
#
```

```
# Here are a few example rules:
```

```
#
```

```
# alert TCP any any -> any 80 (msg: "EXPLOIT ntpdx overflow";
```

```
# dsize: > 128; classtype:attempted-admin; priority:10;
```

```
#
```

```
# alert TCP any any -> any 25 (msg:"SMTP expn root"; flags:A+; \
```

```
# content:"expn root"; nocase; classtype:attempted-recon;)
```

```
#
```

```
# The first rule will set its type to "attempted-admin" and override
```

```
# the default priority for that type to 10.
```

```
#
# The second rule set its type to "attempted-recon" and set its
# priority to the default for that type.
#
#
# config classification:shortname,short description,priority
#
config classification: not-suspicious,Not Suspicious Traffic,3
config classification: unknown,Unknown Traffic,3
config classification: bad-unknown,Potentially Bad Traffic, 2
config classification: attempted-recon,Attempted Information Leak,2
config classification: successful-recon-limited,Information Leak,2
config classification: successful-recon-largescale,Large Scale Information Leak,2
```

Также есть справочный файл, в котором указаны URL-адреса источников информации.

```
[albert@VPN /usr/local/etc/suricata]$ cat reference.config
# config reference: system URL
config reference: bugtraq http://www.securityfocus.com/bid/
config reference: bid http://www.securityfocus.com/bid/
config reference: cve http://cve.mitre.org/cgi-bin/cvename.cgi?name=
#config reference: cve http://cvedetails.com/cve/
config reference: secunia http://www.secunia.com/advisories/
#whitehats is unfortunately gone
config reference: arachNIDS http://www.whitehats.com/info/IDS
config reference: McAfee http://vil.nai.com/vil/content/v_
config reference: nessus http://cgi.nessus.org/plugins/dump.php3?id=
config reference: url http://
```

```
config reference: et http://doc.emergingthreats.net/

.....

.....

[albert@VPN /usr/local/etc/suricata]$
```

Еще один файл, о котором стоит упомянуть, — это threshold.config, в котором можно указать, насколько система обнаружения вторжений должна быть чувствительна, чтобы предупреждать вас о возможных проблемах. Помните, что система обнаружения вторжений может быть довольно шумным элементом в наборе инструментов для защиты вашего офиса, компании или даже двух небольших серверов. Настройка системы требует времени и внимания, так что будьте готовы к ложным срабатываниям, особенно в средних и крупных сетях. Вот фрагмент файла threshold.config:

```
[albert@VPN /usr/local/etc/suricata]$ cat threshold.config

# Thresholding:

#

# This feature is used to reduce the number of logged alerts for noisy rules.

# Thresholding commands limit the number of times a particular event is logged

# during a specified time interval.

#

# The syntax is the following:

#

# threshold gen_id <gen_id>, sig_id <sig_id>, type <limit|threshold|both>, track <by_src|by_dst>, count <n>, seconds <t>

#

# event_filter gen_id <gen_id>, sig_id <sig_id>, type <limit|threshold|both>, track <by_src|by_dst>, count <n>, seconds <t>

#

# suppress gen_id <gid>, sig_id <sid>

# suppress gen_id <gid>, sig_id <sid>, track <by_src|by_dst>, ip <ip|subnet>

#

# The options are documented at https://suricata.readthedocs.io/en/latest/configuration/global-thresholds.html
```

```
#

# Please note that thresholding can also be set inside a signature. The interaction between rule based thresholds

# and global thresholds is documented here:

# https://suricata.readthedocs.io/en/latest/configuration/global-thresholds.html#global-thresholds-vs-rule-thresholds

# Limit to 10 alerts every 10 seconds for each source host

#threshold gen_id 0, sig_id 0, type threshold, track by_src, count 10, seconds 10

# Limit to 1 alert every 10 seconds for signature with sid 2404000

#threshold gen_id 1, sig_id 2404000, type threshold, track by_dst, count 1, seconds 10

# Avoid to alert on f-secure update

# Example taken from https://blog.inliniac.net/2012/03/07/f-secure-av-updates-and-suricata-ips/

#suppress gen_id 1, sig_id 2009557, track by_src, ip 217.110.97.128/25

#suppress gen_id 1, sig_id 2012086, track by_src, ip 217.110.97.128/25

#suppress gen_id 1, sig_id 2003614, track by_src, ip 217.110.97.128/25

[albert@VPN /usr/local/etc/suricata]$
```

Самый важный файл в Suricata — это основной конфигурационный файл suricata.yaml. В файлах .yaml нужно соблюдать отступы. Если их не соблюдать, правила не загрузятся, и файл будет бесполезен. Этот файл находится в папке:

```
/usr/local/etc/suricata/suricata.yaml
```

Файл довольно большой, поэтому приготовьтесь использовать grep с флагом -n, чтобы найти строку, которую нужно настроить. Официальное руководство по администрированию будет вам очень полезно, ведь с помощью Suricata можно сделать очень многое.

Прежде чем мы углубимся в эту тему, давайте в общих чертах рассмотрим, что к чему. Первое, что нужно понять, — какой тип сети мы пытаемся контролировать: глобальную или локальную. По умолчанию в файле suricata.yaml (помните, что это основной файл конфигурации) зарезервированный блок IP-адресов интерпретируется как локальный, как видно из следующего фрагмента. Для локальной сети это нормально, но вам, возможно, придется внести некоторые изменения. Например, если у вас нет SQL-сервера, вы можете отключить эту переменную. Возможно, у вас не сервер OracleDB, а сервер MySQL, поэтому порт 3306 должен быть указан в разделе «port-groups». Просто уделите немного времени изучению следующего снимка из файла suricata.yaml.

```
##
```

```
## Step 1: inform Suricata about your network

##

vars:

# more specific is better for alert accuracy and performance

address-groups:

HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]"

#HOME_NET: "[192.168.0.0/16]"

#HOME_NET: "[10.0.0.0/8]"

#HOME_NET: "[172.16.0.0/12]"

#HOME_NET: "any"

EXTERNAL_NET: "!$HOME_NET"

#EXTERNAL_NET: "any"

HTTP_SERVERS: "$HOME_NET"

SMTP_SERVERS: "$HOME_NET"

SQL_SERVERS: "$HOME_NET"

DNS_SERVERS: "$HOME_NET"

TELNET_SERVERS: "$HOME_NET"

AIM_SERVERS: "$EXTERNAL_NET"

DC_SERVERS: "$HOME_NET"

DNP3_SERVER: "$HOME_NET"

DNP3_CLIENT: "$HOME_NET"

MODBUS_CLIENT: "$HOME_NET"

MODBUS_SERVER: "$HOME_NET"

ENIP_CLIENT: "$HOME_NET"

ENIP_SERVER: "$HOME_NET"

port-groups:

HTTP_PORTS: "80"
```

```
SHELLCODE_PORTS: "!80"

ORACLE_PORTS: 1521

SSH_PORTS: 22

DNP3_PORTS: 20000

MODBUS_PORTS: 502

FILE_DATA_PORTS: "[$HTTP_PORTS,110,143]"

FTP_PORTS: 21

VXLAN_PORTS: 4789

##
```

Вторая важная тема — формат вывода данных, о котором вы можете подробно прочитать по этой ссылке. Если вкратце, то по умолчанию включены строковое оповещение fast и EVE (расширяемый формат событий), а также ряд протоколов, таких как http, snmp, smb, dhcp и многие другие. Настоятельно рекомендуем внимательно изучить документацию, но даже просто просмотрев основной конфигурационный файл, вы поймете, что к чему. Для этого вы всегда можете использовать следующую команду и нажать Enter, чтобы появилась новая строка. Чтобы выйти из программы, просто введите «q».

```
cat /usr/local/etc/suricata/suricata.yaml | less
```

Как уже было сказано, Suricata регистрирует оповещения в нескольких типах журналов, таких как журнал быстрого формата или EVE, а также в журнале suricata.log, где фиксируются и другие типы событий, например проблемы с некоторыми правилами, неполадки в работе демона и т. д. Запомните, где находятся журналы и с чем они связаны.

```
[albert@VPN /usr/local/etc/suricata]$ sudo ls -la /var/log/suricata

total 1582

drwx------ 2 root wheel 6 28 des. 08:24 .

drwxr-xr-x 4 root wheel 52 28 des. 08:00 ..

-rw-r--r-- 1 root wheel 1207757 28 des. 20:42 eve.json

-rw-r--r-- 1 root wheel 0 28 des. 08:24 fast.log

-rw-r--r-- 1 root wheel 643645 28 des. 20:42 stats.log

-rw-r--r-- 1 root wheel 3957 28 des. 20:17 suricata.log

[albert@VPN /usr/local/etc/suricata]$
```

После того как мы настроили Suricata в соответствии со своими предпочтениями, мы можем получить доступ к правилам, проверить, все ли в порядке, и запустить их. Следующая команда также используется для обновления установленных наборов правил.

```
[albert@VPN ~]$ sudo suricata-update

28/12/2019 -- 08:24:27 - <Info> -- Using data-directory /var/lib/suricata.

28/12/2019 -- 08:24:27 - <Info> -- Using Suricata configuration /usr/local/etc/suricata/suricata.yaml

28/12/2019 -- 08:24:27 - <Info> -- Using /usr/local/share/suricata/rules for Suricata provided rules.

28/12/2019 -- 08:24:27 - <Info> -- Found Suricata version 5.0.1 at /usr/local/bin/suricata.

28/12/2019 -- 08:24:27 - <Info> -- Loading /usr/local/etc/suricata/suricata.yaml

28/12/2019 -- 08:24:27 - <Info> -- Disabling rules for protocol modbus

28/12/2019 -- 08:24:27 - <Info> -- Disabling rules for protocol dnp3

28/12/2019 -- 08:24:27 - <Info> -- Disabling rules for protocol enip

28/12/2019 -- 08:24:27 - <Info> -- No sources configured, will use Emerging Threats Open

28/12/2019 -- 08:24:27 - <Info> -- Fetching https://rules.emergingthreats.net/open/suricata-5.0.1/emerging.rules.tar.gz.

100% - 2516963/2516963

28/12/2019 -- 08:24:28 - <Info> -- Done.

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/app-layer-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/decoder-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/dhcp-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/dnp3-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/dns-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/files.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/http-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/ipsec-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/kerberos-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/modbus-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/nfs-events.rules
```

```
28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/ntp-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/smb-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/smtp-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/stream-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Loading distribution rule file /usr/local/share/suricata/rules/tls-events.rules

28/12/2019 -- 08:24:28 - <Info> -- Ignoring file rules/emerging-deleted.rules

28/12/2019 -- 08:24:31 - <Info> -- Loaded 26103 rules.

28/12/2019 -- 08:24:31 - <Warning> -- Disabling ja3 rules as Suricata is built without libnss.

28/12/2019 -- 08:24:31 - <Info> -- 122 ja3_hash rules disabled.

28/12/2019 -- 08:24:31 - <Info> -- Disabled 136 rules.

28/12/2019 -- 08:24:31 - <Info> -- Enabled 0 rules.

28/12/2019 -- 08:24:31 - <Info> -- Modified 0 rules.

28/12/2019 -- 08:24:31 - <Info> -- Dropped 0 rules.

28/12/2019 -- 08:24:31 - <Info> -- Enabled 59 rules for flowbit dependencies.

28/12/2019 -- 08:24:31 - <Info> -- Creating directory /var/lib/suricata/rules.

28/12/2019 -- 08:24:31 - <Info> -- Backing up current rules.

28/12/2019 -- 08:24:31 - <Info> -- Writing rules to /var/lib/suricata/rules/suricata.rules: total: 26103; enabled: 20835; added: 26103; removed 0; modified: 0

28/12/2019 -- 08:24:31 - <Info> -- Testing with suricata -T.

28/12/2019 -- 08:24:43 - <Info> -- Done.

[albert@VPN ~]$
```

Как мы уже видели, команда suricata-update загружает правила и проверяет их. Вы можете автоматизировать этот процесс, запустив специальное задание cron на каждый день. Кстати, обратите внимание, где находятся правила. Все они собраны в одном файле по следующему пути:

```
/var/lib/suricata/rules/suricata.rules
```

Suricata может работать в таком режиме, но управлять правилами таким образом довольно сложно. В другой статье мы расскажем о Oinkmaster — программе для управления правилами Snort (еще одного IDS), которая отлично работает и в Suricata. IDS может быть довольно «шумным» инструментом, и вы можете прийти в отчаяние, когда увидите, сколько

времени уходит на его тонкую настройку. Oinkmaster может помочь в этом, поскольку вы можете отключать отдельные наборы правил, а не целые источники.

Еще одна интересная команда — suricata-update list-sources. Она показывает происхождение правил, например названия компаний, лицензии и некоторые другие параметры.

```
[albert@VPN /usr/local/etc/suricata]$ sudo suricata-update list-sources

28/12/2019 -- 19:50:41 - <Info> -- Using data-directory /var/lib/suricata.

28/12/2019 -- 19:50:41 - <Info> -- Using Suricata configuration /usr/local/etc/suricata/suricata.yaml

28/12/2019 -- 19:50:41 - <Info> -- Using /usr/local/share/suricata/rules for Suricata provided rules.

28/12/2019 -- 19:50:41 - <Info> -- Found Suricata version 5.0.1 at /usr/local/bin/suricata.

28/12/2019 -- 19:50:41 - <Info> -- No source index found, running update-sources

28/12/2019 -- 19:50:41 - <Info> -- Downloading https://www.openinfosecfoundation.org/rules/index.yaml

28/12/2019 -- 19:50:41 - <Info> -- Saved /var/lib/suricata/update/cache/index.yaml

Name: et/open

Vendor: Proofpoint

Summary: Emerging Threats Open Ruleset

License: MIT

Name: et/pro

Vendor: Proofpoint

Summary: Emerging Threats Pro Ruleset

License: Commercial

Replaces: et/open

Parameters: secret-code

Subscription: https://www.proofpoint.com/us/threat-insight/et-pro-ruleset

Name: oisf/trafficid

Vendor: OISF

Summary: Suricata Traffic ID ruleset

License: MIT
```

Name: ptresearch/attackdetection

Vendor: Positive Technologies

Summary: Positive Technologies Attack Detection Team ruleset

License: Custom

Name: scwx/malware

Vendor: Secureworks

Summary: Secureworks suricata-malware ruleset

License: Commercial

Parameters: secret-code

Subscription: https://www.secureworks.com/contact/ (Please reference CTU Countermeasures)

Name: scwx/security

Vendor: Secureworks

Summary: Secureworks suricata-security ruleset

License: Commercial

Parameters: secret-code

Subscription: https://www.secureworks.com/contact/ (Please reference CTU Countermeasures)

Name: sslbl/ssl-fp-blacklist

Vendor: Abuse.ch

Summary: Abuse.ch SSL Blacklist

License: Non-Commercial

Name: sslbl/ja3-fingerprints

Vendor: Abuse.ch

Summary: Abuse.ch Suricata JA3 Fingerprint Ruleset

License: Non-Commercial

Name: etnetera/aggressive

Vendor: Etnetera a.s.

Summary: Etnetera aggressive IP blacklist

```
License: MIT

Name: tgreen/hunting

Vendor: tgreen

Summary: Threat hunting rules

License: GPLv3

[albert@VPN /usr/local/etc/suricata]$
```

Теперь, когда мы кое-что прояснили, можно запускать Suricata без каких-либо изменений в файле suricata.yaml. Очевидно, что в зависимости от типа используемой сети потребуются корректировки. В противном случае вы будете получать слишком много предупреждений о ложных срабатываниях.

Если мы не загрузили правила до запуска Suricata, не волнуйтесь, это можно сделать сразу после запуска.

```
[albert@VPN /usr/local/etc/suricata]$ sudo service suricata start

Starting suricata.

[100403] 28/12/2019 -- 20:53:16 - (suricata.c:1084) <Notice> (LogVersion) -- This is Suricata version 5.0.1 RELEASE running in SYSTEM mode

[albert@VPN /usr/local/etc/suricata]$
```

Теперь проверим, что все работает.

```
[albert@VPN /usr/local/etc/suricata]$ ps aux | grep suricata

root 24659 0,0 38,7 417296 389032 - Ss 20:53 0:17,55 /usr/local/bin/suricata -D --pcap=vtnet0 --pidfile /var/run/suricata.pid -c /usr/local/etc/suricata/suricata.yaml

albert 24683 0,0 0,0 524 336 1 R+ 20:59 0:00,00 grep suricata

[albert@VPN /usr/local/etc/suricata]$
```

А если вам интересно, можете заглянуть в файл suricata.log, чтобы убедиться, что все прошло гладко.

```
[albert@VPN /usr/local/etc/suricata]$ sudo tail /var/log/suricata/suricata.log

[100380] 28/12/2019 -- 20:53:18 - (detect-engine-build.c:1416) <Info> (SigAddressPrepareStage1) -- 20838 signatures processed. 1067 are IP-only rules, 4837 are inspecting packet payload, 14705 inspect application layer, 103 are decoder event only

[100380] 28/12/2019 -- 20:53:28 - (util-runmodes.c:173) <Info> (RunModeSetLiveCaptureAutoFp) -- Using 1 live device(s).
```

```
[100427] 28/12/2019 -- 20:53:28 - (source-pcap.c:351) <Info> (ReceivePcapThreadInit) -- using interface vtnet0

[100427] 28/12/2019 -- 20:53:28 - (source-pcap.c:362) <Info> (ReceivePcapThreadInit) -- running in 'auto' checksum mode. Detection of interface state will require 1000ULL packets

[100427] 28/12/2019 -- 20:53:28 - (util-ioctl.c:112) <Info> (GetIfaceMTU) -- Found an MTU of 1500 for 'vtnet0'

[100427] 28/12/2019 -- 20:53:28 - (source-pcap.c:399) <Info> (ReceivePcapThreadInit) -- Set snaplen to 1524 for 'vtnet0'

[100380] 28/12/2019 -- 20:53:28 - (runmode-pcap.c:295) <Info> (RunModeIdsPcapAutoFp) -- RunModeIdsPcapAutoFp initialised

[100380] 28/12/2019 -- 20:53:28 - (util-conf.c:162) <Info> (ConfUnixSocketIsEnable) -- Running in live mode, activating unix socket

[100380] 28/12/2019 -- 20:53:28 - (unix-manager.c:129) <Info> (UnixNew) -- Using unix socket file '/var/run/suricata/suricata-command.socket'

[100380] 28/12/2019 -- 20:53:28 - (tm-threads.c:2165) <Notice> (TmThreadWaitOnThreadInit) -- all 2 packet processing threads, 4 management threads initialized, engine started.

[albert@VPN /usr/local/etc/suricata]$
```

Как видите, Suricata работает без проблем. В следующих статьях мы расскажем, как запустить службу системного журнала для отправки логов в SIEM, как управлять правилами с помощью Oinkmaster вместо обычных команд и как добавить стек ELK для графического отображения оповещений.

Краткий список того, на что следует обратить внимание при установке Suricata на FreeBSD.

– Не забудьте назначить хотя бы один интерфейс.

– Настройка локальной или глобальной сети.

– По умолчанию включены быстрый формат вывода, EVE и статистика. Активируйте все остальные необходимые параметры.

– Настройка пороговых значений. Границы срабатывания оповещения.

– Настройка классификации. Настройте приоритет любой категории, которую нужно изменить.

– Настройка ссылок. Добавьте источники или отключите те, которые вам не нужны.

– Системный журнал. Если вам нужно отправлять журналы в программное обеспечение SIEM, не забудьте включить это.

– Ротация журналов. Вы можете начать читать это руководство.

– Обновления исходного кода. Задания Cron могут справиться с этим.

Revision #2
Created 8 May 2026 11:15:45 by buzz
Updated 8 May 2026 11:18:47 by buzz