OTUS

Методическое пособие
по выполнению домашнего задания курса
"**Администратор Linux. Professional**"

Стенд с Vagrant с Rsyslog

Содержание

1. Введение

Функция системного журналирования (логирование) – это основной источник информации о работе системы и ошибках.
В системе Linux почти все действия записываются. Именно эти данные помогают разбираться в проблемах с ОС.

Логи могут храниться как локально, так и пересылаться на удаленную систему. Пересылка логов имеет следующие плюсы:

- Возможность централизованного сбора и анализа логов Все логи со всех устройств прилетают в одно место. Это значительно упростит работу с логами.
- Защита от удаления логов на локальной машине.
- Оптимизация места на диске в локальной ОС. Логи не будут храниться в ОС, т.к. будут сразу пересылаться в систему сбора логов. Данная функция настраивается отдельно.

В ОС Linux главным файлом локального журналирования является:
Ubuntu/Debian — /var/log/syslog
RHEL/CentOS — /var/log/messages
Логи в ОС можно настроить.
Например, указывать больше информации или отключить логирование конкретного компонента.

Помимо логов, в Unix-системах используют аудит.
В linux эту функцию выполнеят linux audit daemon.
Linux Audit Daemon – это среда, позволяющая проводить аудит событий в системе Linux. Используя мощную систему аудита возможно отслеживать многие типы событий для мониторинга и проверки системы, например:
доступ к файлам;
изменение прав на файлы;
просмотр пользователей, изменивших конкретный файл;

обнаружение несанкционированных изменений;
мониторинг системных вызовов и функций;
обнаружение аномалий, таких как сбои;
мониторинг набора команд.

Аудит различает 4 вида доступа к файлу:
r — чтение
w — запись в файл
x — выполнение файла
a — изменение атрибута

Для выполнения задания потребуется:

1. ПК на Unix с 8ГБ ОЗУ или виртуальная машина с включенной Nested Virtualization.

Предварительно установленное и настроенное следующее ПО:

1. Hashicorp Vagrant
   https://www.vagrantup.com/downloads
2. Oracle VirtualBox
   https://www.virtualbox.org/wiki/Linux_Downloads
3. Ansible (версия 2.7 и выше)
   https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html

2. Цели домашнего задания

1) Научится проектировать централизованный сбор логов.
2) Рассмотреть особенности разных платформ для сбора логов.

3. Описание домашнего задания

1. В Vagrant разворачиваем 2 виртуальные машины web и log
2. на web настраиваем nginx
3. на log настраиваем центральный лог сервер на любой системе на выбор
journald;
rsyslog;
elk.
4. настраиваем аудит, следящий за изменением конфигов nginx

Все критичные логи с web должны собираться и локально и удаленно.
Все логи с nginx должны уходить на удаленный сервер (локально только критичные).
Логи аудита должны также уходить на удаленную систему.

Формат сдачи ДЗ - vagrant + ansible

Дополнительное задание
развернуть еще машину с elk
таким образом настроить 2 центральных лог системы elk и какую либо еще;
в elk должны уходить только логи нжинкса;
во вторую систему все остальное.

4. Пошаговая инструкция выполнения домашнего задания

Все дальнейшие действия были проверены при использовании Vagrant 2.2.19, VirtualBox v6.1.26 r145957 и образа CentOS 7 (2004.01) из Vagrant cloud. Серьёзные отступления от этой конфигурации могут потребовать адаптации с вашей стороны.

1. Создаём виртуальные машины

Создаём каталог, в котором будут храниться настройки виртуальной машины. В каталоге создаём файл с именем Vagrantfile, добавляем в него следующее содержимое:

Результатом выполнения команды *vagrant up* станут 2 созданные виртуальные машины

Заходим на web-сервер: *vagrant ssh web*

```
Vagrant.configure("2") do |config|
  # Base VM OS configuration.
  config.vm.box = "centos/7"
  config.vm.box_version = "2004.01"

  config.vm.provider :virtualbox do |v|
    v.memory = 512
    v.cpus = 1
  end

  # Define two VMs with static private IP addresses.
  boxes = [
    { :name => "web",
      :ip => "192.168.50.10",
    },
    { :name => "log",
      :ip => "192.168.50.15",
    }
  ]
  # Provision each of the VMs.
  boxes.each do |opts|
    config.vm.define opts[:name] do |config|
      config.vm.hostname = opts[:name]
      config.vm.network "private_network", ip: opts[:ip]

    end
  end
end
```

Дальнейшие действия выполняются от пользователя root. Переходим в root пользователя: sudo -i

Для правильной работы с логами, нужно, чтобы на всех хостах было настроено одинаковое время.

Укажем часовой пояс (Московское время):
cp/usr/share/zoneinfo/Europe/Moscow /etc/localtime

Перезупустим службу NTP Chrony: systemctl restart chronyd

Проверим, что служба работает корректно: systemctl status chronyd

```
[root@web ~]# systemctl restart chronyd
[root@web ~]# systemctl status chronyd
● chronyd.service - NTP client/server
   Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2021-12-12 12:56:51 MSK; 8h ago
     Docs: man:chronyd(8)
           man:chrony.conf(5)
```

Далее проверим, что время и дата указаны правильно: date

```
[root@web ~]# date
Sun Dec 12 21:40:23 MSK 2021
[root@web ~]#
```

Настроить NTP нужно на обоих серверах.

Также, для удобства редактирования конфигурационных файлов можно установить текстовый редактор vim: yum install -y vim

2. Установка nginx на виртуальной машине web

Для установки nginx сначала нужно установить epel-release: yum install epel-release

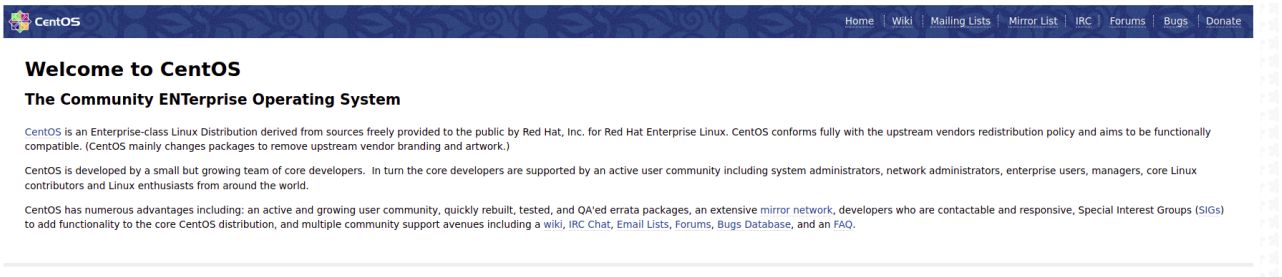
Установим nginx: yum install -y nginx

Проверим, что nginx работает корректно:

```
root@web:~# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2021-12-08 10:00:40 UTC; 17min ago
       Docs: man:nginx(8)
   Main PID: 5986 (nginx)
      Tasks: 3 (limit: 532)
     Memory: 6.5M
     CGroup: /system.slice/nginx.service
             ├─5986 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─5987 nginx: worker process
             └─5988 nginx: worker process

root@web:~#
root@web:~# ss -tln | grep 80
LISTEN   0        511              0.0.0.0:80            0.0.0.0:*
LISTEN   0        511                 [::]:80               [::]:*
root@web:~#
```

Также работу nginx можно проверить на хосте. В браузере ввведем в адерсную строку http://192.168.50.10



Видим что nginx запустился корректно.

3. Настройка центрального сервера сбора логов

Откроем ещё одно окно терминала и подключимся по ssh к ВМ log: vagrant ssh log
Перейдем в пользователя root: sudo -i

rsyslog должен быть установлен по умолчанию в нашёй ОС, проверим это:

```
[root@web ~]# yum list rsyslog
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.awanti.com
 * epel: mirror.yandex.ru
 * extras: mirror.awanti.com
 * updates: mirror.sale-dedic.com
Installed Packages
rsyslog.x86_64
8.24.0-52.el7
@anaconda
Available Packages
rsyslog.x86_64
8.24.0-57.el7_9.1
updates
[root@web ~]#
```

Все настройки Rsyslog хранятся в файле /etc/rsyslog.conf

Для того, чтобы наш сервер мог принимать логи, нам необходимо внести следующие изменения в файл:

Открываем порт 514 (TCP и UDP):

Находим закомментированные строки:

```
# provides UDP syslog reception
#module(load="imudp")
#input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")
```

И приводим их к виду:

```
# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")
```

В конец файла /etc/rsyslog.conf добавляем правила приёма сообщений от хостов:

```
#Add remote logs
$template RemoteLogs,"/var/log/rsyslog/%HOSTNAME%/%PROGRAMNAME%.log"
*.* ?RemoteLogs
& ~
```

Данные параметры будут отправлять в папку /var/log/rsyslog логи, которые будут приходить от других серверов. Например, Access-логи nginx от сервера web, будут идти в файл /var/log/rsyslog**/web/nginx_access.log**

Далее сохраняем файл и перезапускаем службу rsyslog: *systemctl restart rsyslog*

Если ошибок не допущено, то у нас будут видны открытые порты **TCP,UDP 514**:

```
root@log:~# ss -tuln
Netid   State    Recv-Q   Send-Q         Local Address:Port        Peer Address:Port   Process
udp     UNCONN   0        0              0.0.0.0:514              0.0.0.0:*
udp     UNCONN   0        0              127.0.0.53%lo:53         0.0.0.0:*
udp     UNCONN   0        0          10.0.2.15%enp0s3:68          0.0.0.0:*
udp     UNCONN   0        0              [::]:514                 [::]:*
 tcp        LISTEN   0        25             0.0.0.0:514              0.0.0.0:*
 tcp        LISTEN   0        4096           127.0.0.53%lo:53         0.0.0.0:*
 tcp        LISTEN   0        128            0.0.0.0:22               0.0.0.0:*
 tcp        LISTEN   0        25             [::]:514                 [::]:*
 tcp        LISTEN   0        128            [::]:22                  [::]:*
```

```
root@log:~#
```

Далее настроим отправку логов с web-сервера

Заходим на web сервер: *vagrant ssh web*

Переходим в root пользователя: *sudo -i*

Проверим версию nginx: rpm -qa | grep nginx

```
[root@web ~]# rpm -qa | grep nginx
nginx-1.20.1-9.el7.x86_64
nginx-filesystem-1.20.1-9.el7.noarch
[root@web ~]#
```

Версия nginx должна быть 1.7 или выше. В нашем примере используется версия nginx 1.20.

Находим в файле */etc/nginx/nginx.conf* раздел с логами и приводим их к следующему виду:

```
error_log  /var/log/nginx/error.log;
error_log  syslog:server=192.168.50.15:514,tag=nginx_error;
access_log syslog:server=192.168.50.15:514,tag=nginx_access,severity=info combined;
```

Для Access-логов указываем удаленный сервер и уровень логов, которые нужно отправлять. Для error_log добавляем удаленный сервер. Если требуется чтобы логи хранились локально и отправлялись на удаленный сервер, требуется указать 2 строки.

Tag нужен для того, чтобы логи записывались в разные файлы.

По умолчанию, error-логи отправляют логи, которые имеют severity: error, crit, alert и emerg. Если трубуется хранили или пересылать логи с другим severity, то это также можно указать в настройках nginx.

Далее проверяем, что конфигурация nginx указана правильно: nginx -t

```
root@web:/etc/nginx# nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Далее перезапустим nginx: *systemctl restart nginx*

Чтобы проверить, что логи ошибок также улетают на удаленный сервер, можно удалить картинку, к которой будет обращаться nginx во время открытия веб-сраницы:
rm /usr/share/nginx/html/img/header-background.png

Попробуем несколько раз зайти по адресу http://192.168.50.10

Далее заходим на log-сервер и смотрим информацию об nginx:

- *cat /var/log/rsyslog/web/nginx_access.log*
- *cat /var/log/rsyslog/web/nginx_error.log*

```
[root@log ~]# cat /var/log/rsyslog/web/nginx_error.log
Dec 12 22:40:15 web nginx_error: 2021/12/12 22:40:15 [error] 4650#4650: *2 open()
"/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 192.168.50.1,
server: _, request: "GET /favicon.ico HTTP/1.1", host: "192.168.50.10", referrer:
"http://192.168.50.10/"
```

```
[root@log ~]# cat /var/log/rsyslog/web/nginx_access.log
Dec 12 22:37:11 web nginx_access: 192.168.50.1 - - [12/Dec/2021:22:37:11 +0300] "GET / HTTP/1.1"
304 0 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/96.0.4664.93 Safari/537.36"
Dec 12 22:37:13 web nginx_access: 192.168.50.1 - - [12/Dec/2021:22:37:13 +0300] "GET / HTTP/1.1"
304 0 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/96.0.4664.93 Safari/537.36"
```

Видим, что логи отправляются корректно.

4. Настройка аудита, контролирующего изменения конфигурации nginx

За аудит отвечает утилита auditd, в RHEL-based системах обычно он уже предустановлен. Проверим это: rpm -qa | grep audit

```
[root@web ~]# rpm -qa | grep audit
audit-2.8.5-4.el7.x86_64
audit-libs-2.8.5-4.el7.x86_64
[root@web ~]#
```

Настроим аудит изменения конфигурации nginx:

Добавим правило, которое будет отслеживать изменения в конфигруации nginx. Для этого в конец файла /etc/audit/rules.d/audit.rules добавим следующие строки:

```
-w /etc/nginx/nginx.conf -p wa -k nginx_conf
-w /etc/nginx/default.d/ -p wa -k nginx_conf
```

Данные правила позволяют контролировать запись **(w)** и измения атрибутов **(a)** в:

- **/etc/nginx/nginx.conf**
- Всех файлов каталога **/etc/nginx/default.d/**

Для более удобного поиска к событиям добавляется метка **nginx_conf**

Перезапускаем службу auditd: *service auditd restart*

После данных изменений у нас начнут локально записываться логи аудита. Чтобы проверить, что логи аудита начали записываться локально, нужно внести изменения в файл /etc/nginx/nginx.conf *или поменять его атрибут, потом посмотреть информацию об изменениях: ausearch -f /etc/nginx/nginx.conf*

Также можно воспользоваться поиском по файлу /var/log/audit/audit.log, указав наш тэг: *grep nginx_conf /var/log/audit/audit.log*

```
root@web ~]# ausearch -f /etc/nginx/nginx.conf
----
time->Tue Dec 14 14:41:16 2021
node=web type=CONFIG_CHANGE msg=audit(1639482076.543:1600): auid=1000 ses=5 op=updated_rules
path="/etc/nginx/nginx.conf" key="nginx_conf" list=4 res=1
----
time->Tue Dec 14 14:41:16 2021
node=web type=PROCTITLE msg=audit(1639482076.543:1601):
proctitle=76696D002F6574632F6E67696E782F6E67696E782E636F6E66
node=web type=PATH msg=audit(1639482076.543:1601): item=3 name="/etc/nginx/nginx.conf~" inode=11573
dev=08:01 mode=0100644 ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:httpd_config_t:s0 objtype=CREATE
cap_fp=0000000000000000 cap_fi=0000000000000000 cap_fe=0 cap_fver=0
node=web type=PATH msg=audit(1639482076.543:1601): item=2 name="/etc/nginx/nginx.conf" inode=11573
dev=08:01 mode=0100644 ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:httpd_config_t:s0 objtype=DELETE
cap_fp=0000000000000000 cap_fi=0000000000000000 cap_fe=0 cap_fver=0
node=web type=PATH msg=audit(1639482076.543:1601): item=1 name="/etc/nginx/" inode=85 dev=08:01 mode=040755
ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:httpd_config_t:s0 objtype=PARENT cap_fp=0000000000000000
cap_fi=0000000000000000 cap_fe=0 cap_fver=0
node=web type=PATH msg=audit(1639482076.543:1601): item=0 name="/etc/nginx/" inode=85 dev=08:01 mode=040755
ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:httpd_config_t:s0 objtype=PARENT cap_fp=0000000000000000
cap_fi=0000000000000000 cap_fe=0 cap_fver=0
node=web type=CWD msg=audit(1639482076.543:1601):  cwd="/root"
node=web type=SYSCALL msg=audit(1639482076.543:1601): arch=c000003e syscall=82 success=yes exit=0
a0=1ee9170 a1=20bcb60 a2=fffffffffffffe80 a3=7ffed84f05a0 items=4 ppid=4511 pid=27093 auid=1000 uid=0 gid=0
euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0 ses=5 comm="vim" exe="/usr/bin/vim"
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 key="nginx_conf"
----

[root@web ~]# grep nginx_conf /var/log/audit/audit.log
node=web type=CONFIG_CHANGE msg=audit(1639482050.715:1593): auid=4294967295 ses=4294967295
subj=system_u:system_r:unconfined_service_t:s0 op=remove_rule key="nginx_conf" list=4 res=1
node=web type=CONFIG_CHANGE msg=audit(1639482050.715:1594): auid=4294967295 ses=4294967295
subj=system_u:system_r:unconfined_service_t:s0 op=remove_rule key="nginx_conf" list=4 res=1
node=web type=CONFIG_CHANGE msg=audit(1639482050.717:1597): auid=4294967295 ses=4294967295
subj=system_u:system_r:unconfined_service_t:s0 op=add_rule key="nginx_conf" list=4 res=1
```

```
node=web type=CONFIG_CHANGE msg=audit(1639482050.717:1598): auid=4294967295 ses=4294967295
subj=system_u:system_r:unconfined_service_t:s0 op=add_rule key="nginx_conf" list=4 res=1
node=web type=CONFIG_CHANGE msg=audit(1639482076.543:1600): auid=1000 ses=5 op=updated_rules
path="/etc/nginx/nginx.conf" key="nginx_conf" list=4 res=1
node=web type=SYSCALL msg=audit(1639482076.543:1601): arch=c000003e syscall=82 success=yes exit=0
a0=1ee9170 a1=20bcb60 a2=fffffffffffffe80 a3=7ffed84f05a0 items=4 ppid=4511 pid=27093 auid=1000 uid=0 gid=0
euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0 ses=5 comm="vim" exe="/usr/bin/vim"
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 key="nginx_conf"
```

Далее настроим пересылку логов на удаленный сервер. Auditd по умолчанию не умеет пересылать логи, для пересылки на web-сервере потребуется установить пакет **audispd-plugins**: *yum -y install audispd-plugins*

Найдем и поменяем следующие строки в файле /etc/audit/auditd.conf:

```
log_format = RAW
name_format = HOSTNAME
```

В name_format указываем HOSTNAME, чтобы в логах на удаленном сервере отображалось имя хоста.

В файле /etc/audisp/plugins.d/au-remote.conf поменяем параметр active на yes:

```
active = yes
direction = out
path = /sbin/audisp-remote
type = always
#args =
format = string
```

В файле /etc/audisp/audisp-remote.conf требуется указать адрес сервера и порт, на который будут отправляться логи:

```
[root@web ~]# cat /etc/audisp/audisp-remote.conf

remote_server = 192.168.50.15
port = 60
….
```

Далее перезапускаем службу auditd: *service auditd restart*

На этом настройка web-сервера завершена. Далее настроим Log-сервер.

Отроем порт TCP 60, для этого уберем значки комментария в файле /etc/audit/auditd.conf:

```
tcp_listen_port = 60
```

Перезапускаем службу auditd: *service auditd restart*

На этом настройка пересылки логов аудита закончена. Можем попробовать поменять атрибут у файла /etc/nginx/nginx.conf и проверить на log-сервере, что пришла информация об изменении атрибута:

```
[root@web ~]# ls -l /etc/nginx/nginx.conf
-rw-r--r--. 1 root root 1487 Dec 14 14:41 /etc/nginx/nginx.conf
[root@web ~]# chmod +x /etc/nginx/nginx.conf
[root@web ~]# ls -l /etc/nginx/nginx.conf
-rwxr-xr-x. 1 root root 1487 Dec 14 14:41 /etc/nginx/nginx.conf
[root@web ~]#
```

```
[root@log ~]# grep web /var/log/audit/audit.log
node=web type=SYSCALL msg=audit(1639483845.661:1614): arch=c000003e syscall=268 success=yes
exit=0 a0=ffffffffffffff9c a1=1f3b420 a2=1ed a3=7fff13f1e520 items=1 ppid=4511 pid=27277
auid=1000 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0 ses=5 comm="chmod"
exe="/usr/bin/chmod" subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 key="nginx_conf"
node=web type=CWD msg=audit(1639483845.661:1614):  cwd="/root"
node=web type=PATH msg=audit(1639483845.661:1614): item=0 name="/etc/nginx/nginx.conf"
inode=11574 dev=08:01 mode=0100644 ouid=0 ogid=0 rdev=00:00
```

```
obj=system_u:object_r:httpd_config_t:s0 objtype=NORMAL cap_fp=0000000000000000
cap_fi=0000000000000000 cap_fe=0 cap_fver=0
node=web type=PROCTITLE msg=audit(1639483845.661:1614):
proctitle=63686D6F64002B78002F6574632F6E67696E782F6E67696E782E636F6E66
[root@log ~]#
```

5. Критерий оценивания

Статус "Принято" ставится при выполнении следующих условий:

1. Ссылка на репозиторий GitHub.
2. Логи и скриншоты без Vagrantfile
Помимо базового задания рекомендуется сделать данного задание следующим образом:

3. Все настройки сделать с помощью Ansible и добавить запуск Ansible playbook из Vagrantfile.
Для запуска playbook по команде vagrant up достаточно добавить следующую конструкцию в раздел Boxes:

```
boxes.each do |opts|
    config.vm.define opts[:name] do |config|
      config.vm.hostname = opts[:name]
      config.vm.network "private_network", ip: opts[:ip]

      if opts[:name] == boxes.last[:name]
        config.vm.provision "ansible" do |ansible|
          ansible.playbook = "ansible/provision.yml"
          ansible.inventory_path = "ansible/hosts"
          ansible.host_key_checking = "false"
          ansible.limit = "all"
        end
      end

    end
  end
```

4. Предоставить Vagrantfile

Опционально для выполнения:
* развернуть еще машину elk
- таким образом настроить 2 центральных лог системы elk и какую либо еще;
- в elk должны уходить только логи nginx;
- во вторую систему все остальное.

6. Рекомендуемые источники

Статья «Настройка rsyslog для хранения логов на удаленном сервере»
https://www.dmosk.ru/miniinstruktions.php?mini=rsyslog
Статья «Запись в syslog»
https://nginx.org/ru/docs/syslog.html
Статья «Директивы в nginx»
https://nginx.org/ru/docs/ngx_core_module.html#error_log
Статья «How to Setup Rsyslog Client to Send Logs to Rsyslog Server in CentOS 7»
https://www.tecmint.com/setup-rsyslog-client-to-send-logs-to-rsyslog-server-in-centos-7/
Статья «Configure Audit Service to Send Audit Messages to Another Server»
https://www.lisenet.com/2019/configure-audit-service-to-send-audit-messages-to-another-server/
Описание параметров auditd.conf
https://www.opennet.ru/man.shtml?topic=auditd.conf&category=5&russian=0