

Cloud/Container/VM

- [Настройка аудита VMware ESXi и vCenter](#)
- [Kubernetes \(k8s\) via Rsyslog](#)
- [Kubernetes \(k8s\) via webhook](#)
- [Docker via syslog](#)

Настройка аудита VMware ESXi и vCenter

VMware ESXi

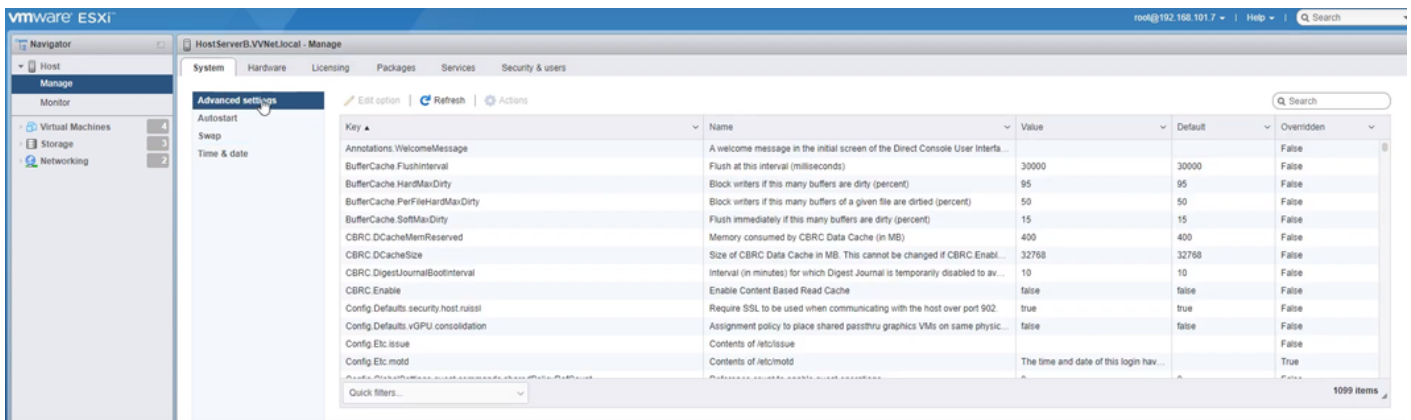
Через веб-интерфейс

Проверьте корректность настроек времени и часового пояса, проверить синхронизацию с NTP-сервером (принять во внимание, что ОС VMware ESXi работает только по UTC).

Выполнить резервное копирование конфигурации ESXi-хоста.

Через web-интерфейс подключиться к ESXi-хосту используя учетную запись root.

В главном меню в разделе навигации развернуть вкладку Host и перейти по пути: **Host - Manage - Advanced Settings**.



В окне поиска набрать **Syslog.global.LogHost**, выбрать параметр Syslog.global.LogHost и отредактировать его.

Edit option | Refresh | Actions

Search

Key	Name	Value	Default	Overridden
Syslog.global.verifySize	Default size of logs before rotation, in KiB. Reset to default on zero.	1024	true	True
Syslog.global.logCheckSSLCerts	Enforce checking of SSL certificates when logging to a remote host.	false	true	True
Syslog.global.logDir	Datastore path of directory to output logs to. Reset to default on null. Example: /scratch/log			True
Syslog.global.logDirUnique	Place logs in a unique subdirectory of logdir, based on hostname.	false	false	False
Syslog.global.logHost	The remote host to output logs to. Reset to default on null. Multiple hosts are supported and must be separated by a space.			False
Syslog.loggers.auth.rotate	Number of rotated logs to keep for this logger. Reset to default on zero.	8	0	True
Syslog.loggers.auth.size	Set size of logs before rotation for this logger, in KiB. Reset to default on zero.	1024	0	True
Syslog.loggers.clomd.rotate	Number of rotated logs to keep for this logger. Reset to default on zero.	8	0	True
Syslog.loggers.clomd.size	Set size of logs before rotation for this logger, in KiB. Reset to default on zero.	1024	0	True
Syslog.loggers.clusterAgent.rotate	Number of rotated logs to keep for this logger. Reset to default on zero.	8	0	True
Syslog.loggers.clusterAgent.size	Set size of logs before rotation for this logger, in KiB. Reset to default on zero.	1024	0	True
Syslog.loggers.cmmmsTimeMachine.rotate	Number of rotated logs to keep for this logger. Reset to default on zero.	8	0	True
Syslog.loggers.cmmmsTimeMachine.size	Set size of logs before rotation for this logger, in KiB. Reset to default on zero.	1024	0	True
Syslog.loggers.cmmmsTimeMachineDump.rotate	Number of rotated logs to keep for this logger. Reset to default on zero.	20	0	True

Quick filters...

1099 items

Укажите протокол, адрес и порт коллектора KUMA и нажмите **Save**.

Edit option - Syslog.global.logHost

New value

udp://192.168.101.212:514
(string)

Save Cancel

Syslog.global.logHost

Description: The remote host to output logs to. Reset to default on null. Multiple hosts are supported and must be separated by a space.

Value: None

Default: None

Далее перейдите на вкладку **Networking - Firewall rules**.

Navigator

Host ServerB.VVNet.local - Networking

Port groups | Virtual switches | Physical NICs | VMkernel NICs | TCP/IP stacks | Firewall rules

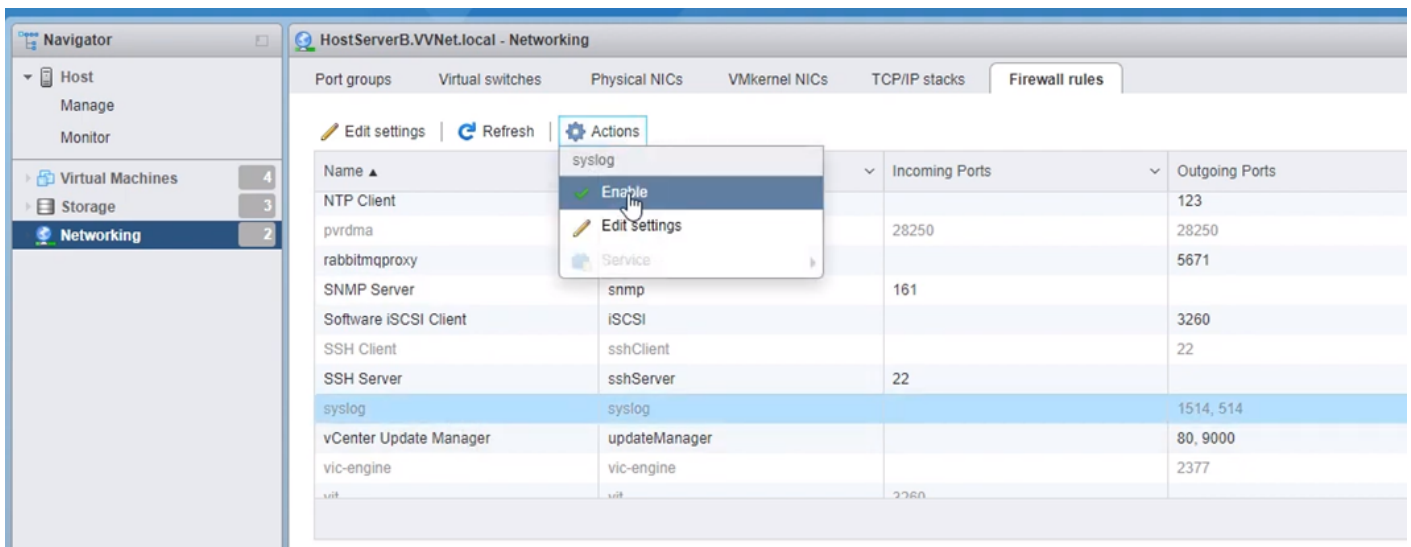
Edit settings | Refresh | Actions

Search

Name	Key	Incoming Ports	Outgoing Ports	Protocols	Service	Daemon
NTP Client	ntpClient		123	UDP	ntpd	Running
pvrdma	pvrdma	28250	28250	TCP	N/A	None
rabbitmqproxy	rabbitmqproxy		5671	TCP	N/A	None
SNMP Server	snmp	161		UDP	snmpd	Stopped
Software iSCSI Client	iSCSI		3260	TCP	N/A	None
SSH Client	sshClient		22	TCP	N/A	None
SSH Server	sshServer	22		TCP	N/A	None
syslog	syslog		1514, 514	UDP, TCP	N/A	None
vCenter Update Manager	updateManager		80, 9000	TCP	N/A	None
vic-engine	vic-engine		2377	TCP	N/A	None

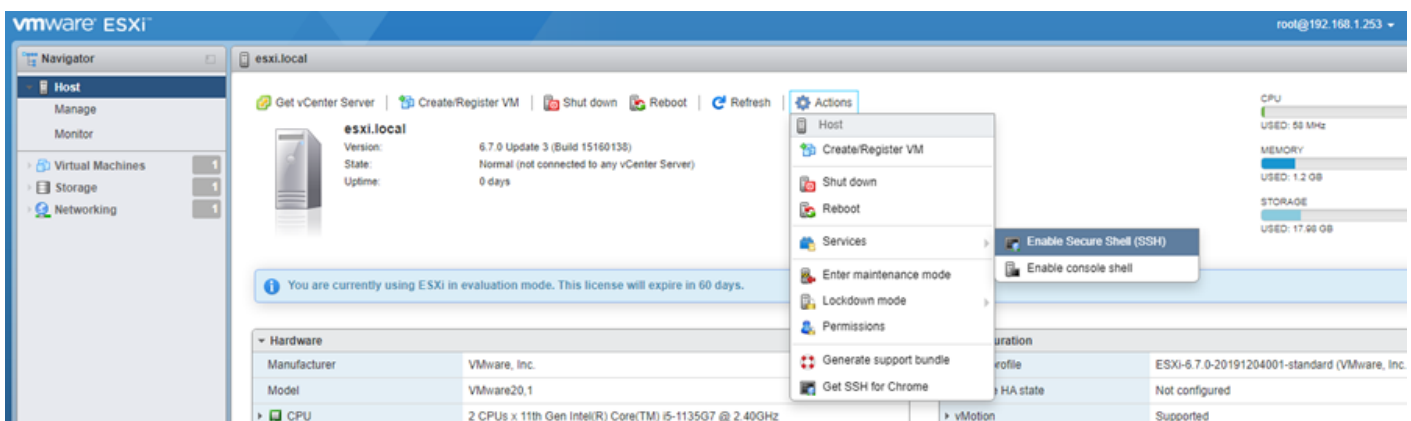
43 items

Найдите правило syslog, выделите его (можно воспользоваться поиском) и включите его, нажав на **Actions - Enable**.



Через SSH

Настройку аудита можно выполнить через SSH. Включить доступ по SSH на ESXi-хосте, перейти по пунктам: **Host - Actions - Services - Enable Secure Shell (SSH)**.



Подключитесь к ESXi-хосту по SSH используя учетную запись root (например через Putty).

- Наберите команду: `esxcli system syslog config set --loghost=udp://192.168.1.250:514` (конфигурирование подключения к syslog-серверу, ip-адрес в команде не является легитимным и указан исключительно для примера).
- Наберите команду: `esxcli network firewall ruleset set --ruleset-id=syslog --enabled=true` (включение разрешающего правила фильтрации для syslog).
- Наберите команду: `esxcli network firewall refresh` (обновление настроек межсетевого экрана ESXi-хоста).
- Наберите команду: `esxcli system syslog config get` (проверка настроек syslog-службы ESXi-хоста):

```
[root@esxi:~] esxcli system syslog config get
Check Certificate Revocation: false
Default Network Retry Timeout: 180
Dropped Log File Rotation Size: 100
Dropped Log File Rotations: 10
Enforce SSLCertificates: true
Local Log Output: /scratch/log
Local Log Output Is Configured: false
Local Log Output Is Persistent: true
Local Logging Default Rotation Size: 1024
Local Logging Default Rotations: 8
Log To Unique Subdirectory: false
Message Queue Drop Mark: 90
Remote Host: udp://192.168.1.250:514
Strict X509Compliance: false
```

- Набрать команду: `esxcli system syslog reload` (перезагрузка syslog-службы ESXi-хоста).
- Авторизоваться на ESXi-хосте, через его web-интерфейс под учетной записью с административными правами и **отключить доступ по SSH** (примечание: Согласно рекомендациям VMware, доступы по SSH и к ESXi-shell нужны только во время диагностических и аварийных работ).

(Опционально) Если необходимо отправлять события **Syslog на другой порт назначения**, необходимо добавить правило для МЭ ESXi, для этого зайдите на хост по SSH.

Создайте файл (используются классические Linux команды) со следующим содержимым, например для порта 5140 (назовем файл syslogPort-5140.xml):

```
<!-- /etc/vmware/firewall/syslogPort-5140.xml -->
<!-- remote syslog configuration -->
<ConfigRoot>
  <service>
    <id>syslogPort-5140</id>
    <rule id='0000'>
      <direction>outbound</direction>
      <protocol>udp</protocol>
      <porttype>dst</porttype>
      <port>5140</port>
    </rule>

    <rule id='0001'>
      <direction>outbound</direction>
```

```
<protocol>tcp</protocol>
<porttype>dst</porttype>
<port>5140</port>
</rule>

<enabled>>false</enabled>
<required>>false</required>
</service>
</ConfigRoot>
```

Для использования этого правила выполните команды ниже, и активируйте его:

- `cp syslogPort-5140.xml /etc/vmware/firewall/`
- `esxcli network firewall unload`
- `esxcli network firewall load`

VMware vCenter

Сделайте snapshot или выполните резервное копирование vCenter.

Через web-браузер подключитесь к vCenter Server Appliance Management Interface (VAMI) используя административную учетную запись (например, administrator@vsphere.local).

Наберите в web-браузере: `https://vcenter.test.local:5480` и ввести административные учетные данные (имя `vcenter.test.local` не является легитимным и указан для примера).

Убедитесь в корректности настроек времени в разделе Time (часовой пояс указан в качестве примера, а тип синхронизации в «Филиале» будет индивидуально зависеть от указанных местных настроек).

Time zone	
Time zone	Europe/Samara
Time synchronization	
Mode	Host
Current appliance time	Wed 03-29-2023 11:42 AM +04

Перейти в раздел **Syslog**, чтобы настроить Forwarding. Нажать кнопку **CONFIGURE**, укажите протокол, адрес и порт коллектора KUMA и нажмите **Save**.

Create Forwarding Configuration

Specify forwarding configuration for remote syslog servers (no more than three).

Server Address	Protocol	Port
192.168.1.250	UDP	514

+ ADD

CANCEL

SAVE

Forwarding Configuration ⓘ

EDIT SEND TEST MESSAGE DELETE

Remote Syslog Host	Protocol	Port	Connection Status
192.168.1.250	UDP	514	Unknown

Отправьте тестовое сообщение:

Send Test Message

Manually verify from remote syslog servers if the message has been received.

Test message: This is a diagnostic syslog test message from vCenter Server.

Servers: 192.168.1.250

CANCEL

SEND

Send Test Message

✓ Successfully sent the test message to all syslog servers.

Manually verify from remote syslog servers if the message has been received.

Test message: This is a diagnostic syslog test message from vCenter Server.

Servers: 192.168.1.250

CANCEL

SEND

Kubernetes (k8s) via Rsyslog

Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

Общее

Настройка логирования Kubernetes (k8s) выполняется путем модификации kube-apiserver.

Подробное описание механизма аудита k8s приведено на официальном [сайте](#). Данная инструкция предназначена для настройки аудита k8s для последующей передачи логов в KUMA.

Настройка k8s

1. Необходимо подключиться к ноде k8s с ролью control plane
2. На ноде создаем директорию, куда будет помещена политика аудита

```
sudo mkdir /etc/kubernetes/audit/
```

3. В созданной директории создаем файл с политикой аудита `/etc/kubernetes/audit/audit-policy.yaml` любым удобным способом. Содержимое файла может варьироваться от целей логирования, ниже приведен пример политики с официального сайта.

Будьте внимательны, конфигурации в k8s как правило задаются в виде файлов YAML, которые чувствительны к отступам. Валидируйте файлы перед их применением во избежание ошибок.

Пример политики аудита k8s

```
apiVersion: audit.k8s.io/v1 # This is required.
kind: Policy
# Don't generate audit events for all requests in RequestReceived stage.
omitStages:
```


- "RequestReceived"

rules:

Log pod changes at RequestResponse level

- level: RequestResponse

resources:

- group: ""

Resource "pods" doesn't match requests to any subresource of pods,

which is consistent with the RBAC policy.

resources: ["pods"]

Log "pods/log", "pods/status" at Metadata level

- level: Metadata

resources:

- group: ""

resources: ["pods/log", "pods/status"]

Don't log requests to a configmap called "controller-leader"

- level: None

resources:

- group: ""

resources: ["configmaps"]

resourceNames: ["controller-leader"]

Don't log watch requests by the "system:kube-proxy" on endpoints or services

- level: None

users: ["system:kube-proxy"]

verbs: ["watch"]

resources:

- group: "" # core API group

resources: ["endpoints", "services"]

Don't log authenticated requests to certain non-resource URL paths.

- level: None

userGroups: ["system:authenticated"]

nonResourceURLs:

- "/api*" # Wildcard matching.

- "/version"

Log the request body of configmap changes in kube-system.

- level: Request

```

resources:
- group: "" # core API group
  resources: ["configmaps"]
# This rule only applies to resources in the "kube-system" namespace.
# The empty string "" can be used to select non-namespaced resources.
namespaces: ["kube-system"]

# Log configmap and secret changes in all other namespaces at the Metadata level.
- level: Metadata
  resources:
  - group: "" # core API group
    resources: ["secrets", "configmaps"]

# Log all other resources in core and extensions at the Request level.
- level: Request
  resources:
  - group: "" # core API group
  - group: "extensions" # Version of group should NOT be included.

# A catch-all rule to log all other requests at the Metadata level.
- level: Metadata
  # Long-running requests like watches that fall under this rule will not
  # generate an audit event in RequestReceived.
  omitStages:
  - "RequestReceived"

```

4. Далее создаем директорию, в которую будут записаны логи аудита k8s

```
sudo mkdir -p /var/log/kubernetes/audit/
```

5. Далее необходимо будет внести изменения в конфигурацию пода kube-apiserver. Перед этим настоятельно рекомендуется сделать резервную копию конфигурации, например, следующей командой из вашей рабочей директории:

```
sudo cp /etc/kubernetes/manifests/kube-apiserver.yaml .
```

6. Вносим изменение в kube-apiserver с помощью команды:

```
sudo vi /etc/kubernetes/manifests/kube-apiserver.yaml
```

7. В секции `spec.containers.command` указываем следующие флаги, соблюдая отступы:

```
- --audit-policy-file=/etc/kubernetes/audit/audit-policy.yaml
- --audit-log-path=/var/log/kubernetes/audit/audit.log
```

Где `/etc/kubernetes/audit/audit-policy.yaml` - путь к политике аудита, а `/var/log/kubernetes/audit/audit.log` - путь к файлу для записи логов.

8. Дополнительно можно определить другие параметры логирования, такие как размер файла логов и количество файлов (подробное описание параметров можно найти [тут](#)):

```
- --audit-log-maxsize=500
- --audit-log-maxbackup=3
```

Настройка секции `spec.containers.command`

```
spec:
  containers:
  - command:
    - kube-apiserver
    - --advertise-address=10.0.2.15
    - --allow-privileged=true
    - --authorization-mode=Node,RBAC
    - --client-ca-file=/etc/kubernetes/pki/ca.crt
    - --enable-admission-plugins=NodeRestriction
    - --enable-bootstrap-token-auth=true
    - --etcd-cafile=/etc/kubernetes/pki/etcd/ca.crt
    - --etcd-certfile=/etc/kubernetes/pki/apiserver-etcd-client.crt
    - --etcd-keyfile=/etc/kubernetes/pki/apiserver-etcd-client.key
    - --etcd-servers=https://127.0.0.1:2379
    - --kubelet-client-certificate=/etc/kubernetes/pki/apiserver-kubelet-client.crt
    - --kubelet-client-key=/etc/kubernetes/pki/apiserver-kubelet-client.key
    - --kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname
    - --proxy-client-cert-file=/etc/kubernetes/pki/front-proxy-client.crt
    - --proxy-client-key-file=/etc/kubernetes/pki/front-proxy-client.key
    - --requestheader-allowed-names=front-proxy-client
    - --requestheader-client-ca-file=/etc/kubernetes/pki/front-proxy-ca.crt
    - --requestheader-extra-headers-prefix=X-Remote-Extra-
    - --requestheader-group-headers=X-Remote-Group
    - --requestheader-username-headers=X-Remote-User
    - --secure-port=6443
    - --service-account-issuer=https://kubernetes.default.svc.cluster.local
    - --service-account-key-file=/etc/kubernetes/pki/sa.pub
    - --service-account-signing-key-file=/etc/kubernetes/pki/sa.key
    - --service-cluster-ip-range=10.96.0.0/12
    - --tls-cert-file=/etc/kubernetes/pki/apiserver.crt
    - --tls-private-key-file=/etc/kubernetes/pki/apiserver.key
    - --audit-policy-file=/etc/kubernetes/audit/policy.yaml
    - --audit-log-path=/etc/kubernetes/audit/audit.log
    - --audit-log-maxsize=500
    - --audit-log-maxbackup=3
    image: registry.k8s.io/kube-apiserver:v1.30.2
    imagePullPolicy: IfNotPresent
    livenessProbe:
```

9. Далее в том же файле создаем соответствующие тома и точки монтирования для политики и директории для хранения логов

10. В секцию `volumes` добавляем следующее соблюдая отступы:

```
- hostPath:
  path: /etc/kubernetes/audit/audit-policy.yaml
  type: File
name: audit
- hostPath:
  path: /var/log/kubernetes/audit/
  type: DirectoryOrCreate
name: audit-log
```

Здесь `/etc/kubernetes/audit/audit-policy.yaml` - путь к политике аудита, а `/var/log/kubernetes/audit/audit.log` - путь к файлу для записи логов.

Настройка секции `volumes`

```

volumes:
- hostPath:
    path: /etc/ssl/certs
    type: DirectoryOrCreate
    name: ca-certs
- hostPath:
    path: /etc/ca-certificates
    type: DirectoryOrCreate
    name: etc-ca-certificates
- hostPath:
    path: /etc/kubernetes/pki
    type: DirectoryOrCreate
    name: k8s-certs
- hostPath:
    path: /usr/local/share/ca-certificates
    type: DirectoryOrCreate
    name: usr-local-share-ca-certificates
- hostPath:
    path: /usr/share/ca-certificates
    type: DirectoryOrCreate
    name: usr-share-ca-certificates
- hostPath:
    path: /etc/kubernetes/audit/audit-policy.yaml
    type: File
    name: audit
- hostPath:
    path: /var/log/kubernetes/audit/
    type: DirectoryOrCreate
    name: audit-log

```

11. В секцию `volumeMounts` добавляем следующее соблюдая отступы:

```

- mountPath: /etc/kubernetes/audit/audit-policy.yaml
  name: audit
  readOnly: true
- mountPath: /var/log/kubernetes/audit/
  name: audit-log
  readOnly: false

```

Настройка секции `volumeMounts`

```
volumeMounts:
- mountPath: /etc/ssl/certs
  name: ca-certs
  readOnly: true
- mountPath: /etc/ca-certificates
  name: etc-ca-certificates
  readOnly: true
- mountPath: /etc/kubernetes/pki
  name: k8s-certs
  readOnly: true
- mountPath: /usr/local/share/ca-certificates
  name: usr-local-share-ca-certificates
  readOnly: true
- mountPath: /usr/share/ca-certificates
  name: usr-share-ca-certificates
  readOnly: true
- mountPath: /etc/kubernetes/audit/audit-policy.yaml
  name: audit
  readOnly: true
- mountPath: /var/log/kubernetes/audit/
  name: audit-log
  readOnly: false
```

12. Сохраняем все внесенные в файл изменения.

Т.к. была изменена конфигурация kube-apiserver, то под будет пересоздан, что может потребовать примерно до 1 минуты времени. Если под не смог подняться, необходимо проверить все внесенные изменения на предмет ошибок и опечаток, а также изучить логи по пути `/var/log/pods/`

Если все было сделано правильно, то под kube-apiserver поднимется и в директории `/var/log/kubernetes/audit/` появится файл `audit.log` и начнет наполняться логами k8s.

Настройка Rsyslog

Настройки ниже приведены для deb-систем.

1. Установка rsyslog

```
apt install rsyslog
```

2. Включение и запуск службы rsyslog

```
systemctl enable rsyslog.service  
systemctl start rsyslog.service
```

3. Создание файла конфигурации для отправки через rsyslog файла лога k8s

```
nano /etc/rsyslog.d/k8s.conf
```

Пример содержимого файла с отправкой по TCP:

```
$ModLoad imfile  
$InputFileName /var/log/kubernetes/audit/audit.log  
$InputFileTag tag_k8s_log:  
$InputFileStateFile k8s_log  
$InputFileSeverity info  
$InputFileFacility local6  
$InputRunFileMonitor  
  
local6.* @@10.10.10.10:7777
```

Где 10.10.10.10 - адрес коллектора KUMA, 7777 - порт коллектора KUMA

Для отправки событий по протоколу UDP последнюю строчку следует заменить на:

```
local6.* @10.10.10.10:7777
```

4. После сохранения изменений в файле необходимо перезапустить сервис Rsyslog командой:

```
systemctl restart rsyslog.service
```

Настройка коллектора KUMA

После того как параметры передачи событий настроены, требуется создать коллектор в веб-интерфейсе KUMA для событий `k8s`.

1. На шаге **Транспорт** укажите тип и порт в соответствии с настройками на стороне *Kubernetes*.

2. На шаге **Парсинг** событий выберите нормализатор **k8s via syslog**.

3. На шаге **Маршрутизация** проверьте, что в набор ресурсов коллектора добавлены следующие точки назначения:

- **Хранилище**. Для отправки обработанных событий в хранилище.

- **Коррелятор**. Для отправки обработанных событий в коррелятор.

Если точки назначения **Хранилище** и **Коррелятор** не добавлены, создайте их.

4. На шаге **Проверка параметров** нажмите **Сохранить и создать сервис**.

5. Скопируйте появившуюся команду для установки коллектора KUMA.

Полезные ссылки

Пакет контента для k8s: https://github.com/KUMA-Community/kuma_content/tree/main/rules/app/kubernetes

Документация по настройке аудита k8s: <https://kubernetes.io/docs/tasks/debug/debug-cluster/audit/>

Kubernetes (k8s) via webhook

Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

Данный способ является экспериментальным. Рекомендуемый способ приведен в статье [k8s via rsyslog](#)

Общее

Настройка логирования Kubernetes (k8s) выполняется путем модификации kube-apiserver. Подробное описание механизма аудита k8s приведено на официальном [сайте](#). Данная инструкция предназначена для настройки аудита k8s для последующей передачи логов в KUMA.

Настройка k8s

1. Необходимо подключиться к ноду k8s с ролью control plane
2. На ноду создаем директорию, куда будет помещена политика аудита

```
sudo mkdir /etc/kubernetes/audit/
```

3. В созданной директории создаем файл с политикой аудита `/etc/kubernetes/audit/audit-policy.yaml` любым удобным способом. Содержимое файла может варьироваться от целей логирования, ниже приведен пример политики с официального сайта.

Будьте внимательны, конфигурации в k8s как правило задаются в виде файлов YAML, которые чувствительны к отступам. Валидируйте файлы перед их применением во избежание ошибок.

Пример политики аудита k8s

```
apiVersion: audit.k8s.io/v1 # This is required.
kind: Policy

# Don't generate audit events for all requests in RequestReceived stage.
omitStages:
  - "RequestReceived"

rules:
  # Log pod changes at RequestResponse level
  - level: RequestResponse
    resources:
      - group: ""
        # Resource "pods" doesn't match requests to any subresource of pods,
        # which is consistent with the RBAC policy.
        resources: ["pods"]
  # Log "pods/log", "pods/status" at Metadata level
  - level: Metadata
    resources:
      - group: ""
        resources: ["pods/log", "pods/status"]

  # Don't log requests to a configmap called "controller-leader"
  - level: None
    resources:
      - group: ""
        resources: ["configmaps"]
        resourceNames: ["controller-leader"]

  # Don't log watch requests by the "system:kube-proxy" on endpoints or services
  - level: None
    users: ["system:kube-proxy"]
    verbs: ["watch"]
    resources:
      - group: "" # core API group
        resources: ["endpoints", "services"]

  # Don't log authenticated requests to certain non-resource URL paths.
  - level: None
    userGroups: ["system:authenticated"]
```

```
nonResourceURLs:
```

- "/api*" # Wildcard matching.
- "/version"

```
# Log the request body of configmap changes in kube-system.
```

```
- level: Request
```

```
resources:
```

```
- group: "" # core API group
```

```
resources: ["configmaps"]
```

```
# This rule only applies to resources in the "kube-system" namespace.
```

```
# The empty string "" can be used to select non-namespaced resources.
```

```
namespaces: ["kube-system"]
```

```
# Log configmap and secret changes in all other namespaces at the Metadata level.
```

```
- level: Metadata
```

```
resources:
```

```
- group: "" # core API group
```

```
resources: ["secrets", "configmaps"]
```

```
# Log all other resources in core and extensions at the Request level.
```

```
- level: Request
```

```
resources:
```

```
- group: "" # core API group
```

```
- group: "extensions" # Version of group should NOT be included.
```

```
# A catch-all rule to log all other requests at the Metadata level.
```

```
- level: Metadata
```

```
# Long-running requests like watches that fall under this rule will not
```

```
# generate an audit event in RequestReceived.
```

```
omitStages:
```

```
- "RequestReceived"
```

4. В этой же директории создаем файл `/etc/kubernetes/audit/audit-webhook.yaml` любым удобным способом. Пример содержимого файла приведен ниже:

```
apiVersion: v1
```

```
kind: Config
```

```
preferences: {}
```

```
clusters:
```

```
- name: kube-auditing
cluster:
  server: http://10.10.10.10:7777/input
users: []
contexts:
- name: default-context
context:
  cluster: kube-auditing
  user: ""
current-context: default-context
```

Где 10.10.10.10 - адрес коллектора KUMA, а 7777 - порт коллектора. Все прочие параметры из примера можно оставлять без изменений.5. Далее создаем директорию, в которую будут записаны логи аудита k8s

5. Далее необходимо будет внести изменения в конфигурацию пода kube-apiserver. Перед этим настоятельно рекомендуется сделать резервную копию конфигурации, например, следующей командой из вашей рабочей директории:

```
sudo cp /etc/kubernetes/manifests/kube-apiserver.yaml .
```

6. Вносим изменение в kube-apiserver с помощью команды:

```
sudo vi /etc/kubernetes/manifests/kube-apiserver.yaml
```

7. В секции `spec.containers.command` указываем следующие флаги, соблюдая отступы:

```
- --audit-webhook-config-file=/etc/kubernetes/audit/audit-webhook.yaml
- --audit-webhook-mode=batch
- --audit-webhook-batch-max-size=1
```

Где `/etc/kubernetes/audit/audit-policy.yaml` - путь к политике аудита.

Настройка секции `spec.containers.command`

```

spec:
  containers:
  - command:
    - kube-apiserver
    - --advertise-address=10.68.85.163
    - --allow-privileged=true
    - --authorization-mode=Node,RBAC
    - --client-ca-file=/etc/kubernetes/pki/ca.crt
    - --enable-admission-plugins=NodeRestriction
    - --enable-bootstrap-token-auth=true
    - --etcd-cafile=/etc/kubernetes/pki/etcd/ca.crt
    - --etcd-certfile=/etc/kubernetes/pki/apiserver-etcd-client.crt
    - --etcd-keyfile=/etc/kubernetes/pki/apiserver-etcd-client.key
    - --etcd-servers=https://127.0.0.1:2379
    - --kubelet-client-certificate=/etc/kubernetes/pki/apiserver-kubelet-client.crt
    - --kubelet-client-key=/etc/kubernetes/pki/apiserver-kubelet-client.key
    - --kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname
    - --proxy-client-cert-file=/etc/kubernetes/pki/front-proxy-client.crt
    - --proxy-client-key-file=/etc/kubernetes/pki/front-proxy-client.key
    - --requestheader-allowed-names=front-proxy-client
    - --requestheader-client-ca-file=/etc/kubernetes/pki/front-proxy-ca.crt
    - --requestheader-extra-headers-prefix=X-Remote-Extra-
    - --requestheader-group-headers=X-Remote-Group
    - --requestheader-username-headers=X-Remote-User
    - --secure-port=6443
    - --service-account-issuer=https://kubernetes.default.svc.cluster.local
    - --service-account-key-file=/etc/kubernetes/pki/sa.pub
    - --service-account-signing-key-file=/etc/kubernetes/pki/sa.key
    - --service-cluster-ip-range=10.96.0.0/12
    - --tls-cert-file=/etc/kubernetes/pki/apiserver.crt
    - --tls-private-key-file=/etc/kubernetes/pki/apiserver.key
    - --audit-policy-file=/etc/kubernetes/audit/audit-policy.yaml
    - --audit-webhook-config-file=/etc/kubernetes/audit/audit-webhook.yaml
    - --audit-webhook-mode=batch
    - --audit-webhook-batch-max-size=1
  image: registry.k8s.io/kube-apiserver:v1.31.1

```

9. Далее в том же файле создаем соответствующий том и точку монтирования для политик

10. В секцию `volumes` добавляем следующее соблюдая отступы:

```

- hostPath:
  path: /etc/kubernetes/audit/
  type: DirectoryOrCreate
  name: k8s-audit

```

Здесь `/etc/kubernetes/audit/` - путь к директории с политикой аудита и настройками webhook

Настройка секции volumes

```
type: DirectoryOrCreate
volumes:
- hostPath:
    path: /etc/ssl/certs
    type: DirectoryOrCreate
    name: ca-certs
- hostPath:
    path: /etc/ca-certificates
    type: DirectoryOrCreate
    name: etc-ca-certificates
- hostPath:
    path: /etc/kubernetes/pki
    type: DirectoryOrCreate
    name: k8s-certs
- hostPath:
    path: /usr/local/share/ca-certificates
    type: DirectoryOrCreate
    name: usr-local-share-ca-certificates
- hostPath:
    path: /usr/share/ca-certificates
    type: DirectoryOrCreate
    name: usr-share-ca-certificates
- hostPath:
    path: /etc/kubernetes/audit/
    type: DirectoryOrCreate
    name: k8s-audit
status: {}
```

11. В секцию `volumeMounts` добавляем следующее соблюдая отступы:

```
- mountPath: /etc/kubernetes/audit/
  name: k8s-audit
  readOnly: true
```

Настройка секции `volumeMounts`

```
volumeMounts:
- mountPath: /etc/ssl/certs
  name: ca-certs
  readOnly: true
- mountPath: /etc/ca-certificates
  name: etc-ca-certificates
  readOnly: true
- mountPath: /etc/kubernetes/pki
  name: k8s-certs
  readOnly: true
- mountPath: /usr/local/share/ca-certificates
  name: usr-local-share-ca-certificates
  readOnly: true
- mountPath: /usr/share/ca-certificates
  name: usr-share-ca-certificates
  readOnly: true
- mountPath: /etc/kubernetes/audit/
  name: k8s-audit
  readOnly: true
```

12. Сохраняем все внесенные в файл изменения.

Т.к. была изменена конфигурация kube-apiserver, то под будет пересоздан, что может потребовать примерно до 1 минуты времени. Если под не смог подняться, необходимо проверить все внесенные изменения на предмет ошибок и опечаток, а также изучить логи по пути `/var/log/pods/`

Настройка коллектора KUMA

После того как параметры передачи событий настроены, требуется создать коллектор в веб-интерфейсе KUMA для событий `k8s`.

1. На шаге **Транспорт** укажите тип и порт в соответствии с настройками на стороне *Kubernetes* (`audit-webhook.yaml`).

2. На шаге **Парсинг** событий выберите нормализатор **k8s via webhook**.

3. На шаге **Маршрутизация** проверьте, что в набор ресурсов коллектора добавлены следующие точки назначения:

- **Хранилище**. Для отправки обработанных событий в хранилище.
- **Коррелятор**. Для отправки обработанных событий в коррелятор.

Если точки назначения **Хранилище** и **Коррелятор** не добавлены, создайте их.

4. На шаге **Проверка параметров** нажмите **Сохранить и создать сервис**.

5. Скопируйте появившуюся команду для установки коллектора KUMA.

Полезные ссылки

Пакет контента для k8s: https://github.com/KUMA-Community/kuma_content/tree/main/rules/app/kubernetes

Документация по настройке аудита k8s: <https://kubernetes.io/docs/tasks/debug/debug-cluster/audit/>

Docker via syslog

Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

Общее

Настройка логирования Docker выполняется путем модификации `/etc/docker/daemon.json`, либо точно для каждого контейнера через указание соответствующих параметров при запуске. В данной статье будет рассмотрен первый вариант.

Настройка Docker

1. Необходимо подключиться к ноде Docker
2. На ноде создать файл конфигурации `/etc/docker/daemon.json`, либо внести изменения в существующий. Пример файла конфигурации представлен ниже.

```
{
  "log-driver": "syslog",
  "log-opts": {
    "syslog-address": "tcp://10.10.10.10:6688",
    "tag": "{{.ID}}/{{.Name}}",
    "syslog-format": "rfc5424"
  }
}
```

Где, `10.10.10.10` - адрес коллектора KUMA, `66888` - порт коллектора KUMA. При необходимости можно также переопределить протокол передачи, формат логов и тегирование. Подробности см. по ссылке в конце статьи.

3. После внесения изменения в файл необходимо перезапустить службу Docker'a:

```
systemctl restart docker.service
```

В результате внесенных изменений события от Docker будут перенаправляться на сервис коллектора KUMA в соответствии с указанными параметрами.

Альтернативно можно настроить логирование на стороне Docker в файл и перенаправлять его содержимое через rsyslog или путем монтирования папки/установки агента KUMA.

Настройка коллектора KUMA

После того как параметры передачи событий настроены, требуется создать коллектор в веб-интерфейсе KUMA для событий `Docker`.

1. На шаге **Транспорт** укажите тип и порт в соответствии с настройками на стороне *Docker*.
2. На шаге **Парсинг** событий выберите нормализатор **[OOTB] Syslog**.

В дальнейшем можно использовать кастомные парсеры в зависимости от приложений, работающих в контейнерах Docker.

3. На шаге **Маршрутизация** проверьте, что в набор ресурсов коллектора добавлены следующие точки назначения:

- **Хранилище**. Для отправки обработанных событий в хранилище.

- **Коррелятор**. Для отправки обработанных событий в коррелятор.

Если точки назначения **Хранилище** и **Коррелятор** не добавлены, создайте их.

4. На шаге **Проверка параметров** нажмите **Сохранить и создать сервис**.

5. Скопируйте появившуюся команду для установки коллектора KUMA.

Полезные ссылки

Документация по настройке syslog в Docker:

<https://docs.docker.com/engine/logging/drivers/syslog/>