

Модуль 2. Сканирование на уязвимости. Сетевые сканирования (vo_HW)

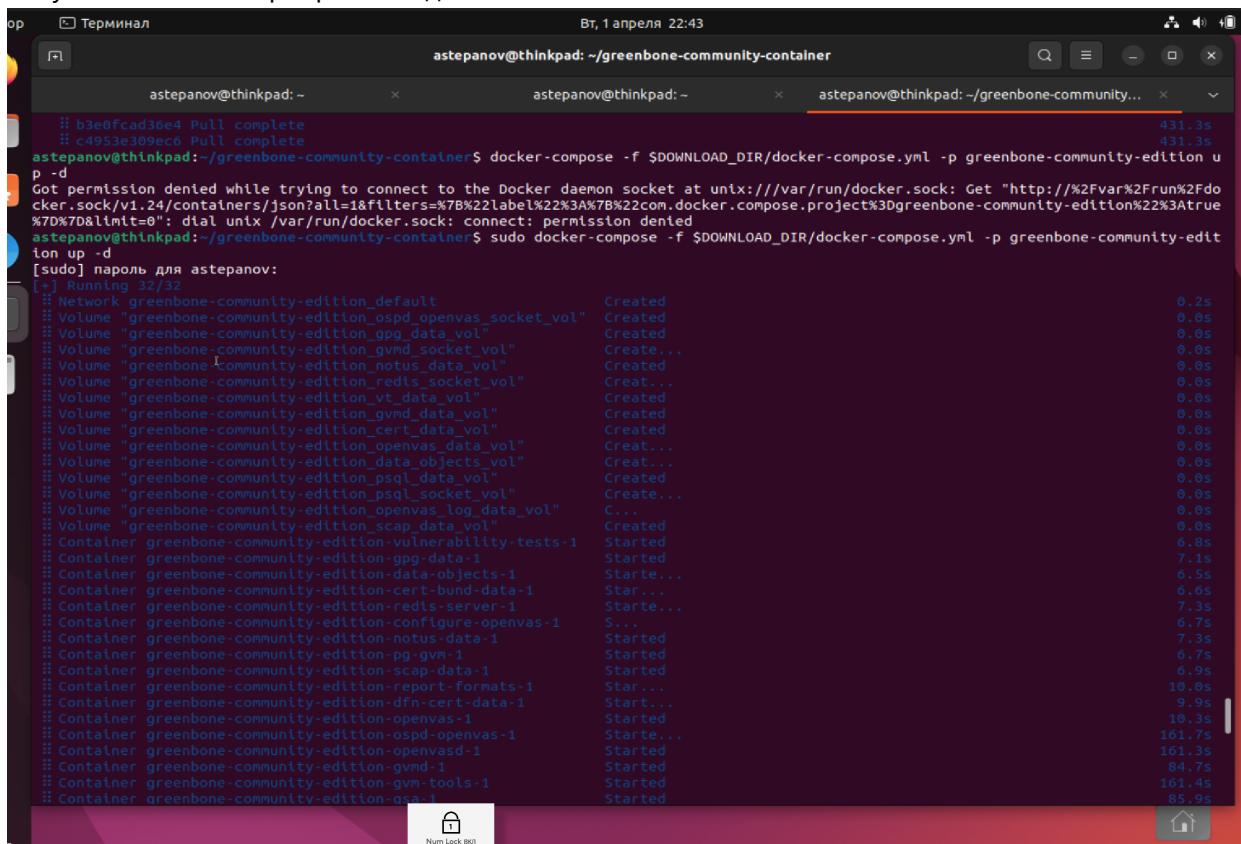
Задание №2. Сканирование с помощью OpenVAS

Шаг 1. Установка OpenVAS.

Устанавливаем OpenVas на виртуальной машине Ubuntu, следуя инструкции из урока по OpenVas в данном модуле

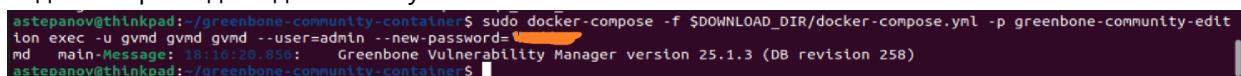
1. Устанавливаем **docker**, **docker-compose**, скачиваем нужный docker-контейнер (**greenbone-community-edition**)

2. Запускаем контейнер в режиме демона



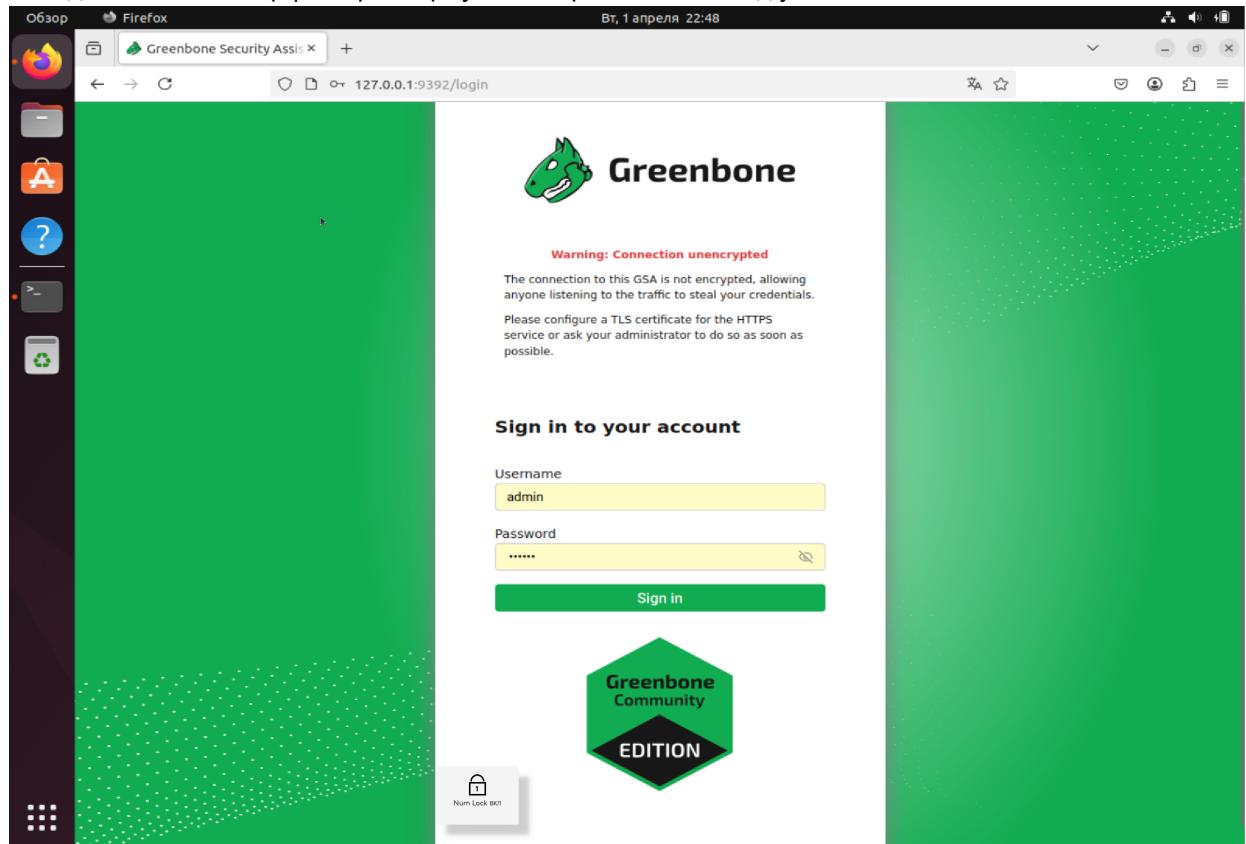
```
astepanov@thinkpad:~/greenbone-community-container$ docker-compose -f $DOWNLOAD_DIR/docker-compose.yml -p greenbone-community-edition up -d
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://<2>var<2>run<2>Fdo
cker.sock/v1.24/containers/json?all=1&filters=%7B%22label%22%3A%7B%22com.docker.compose.project%3Dgreenbone-community-edition%22%3Atrue
%7D%7D&limit=0": dial unix /var/run/docker.sock: connect: permission denied
astepanov@thinkpad:~/greenbone-community-container$ sudo docker-compose -f $DOWNLOAD_DIR/docker-compose.yml -p greenbone-community-edition up -d
[sudo] пароль для astepanov:
[+] Running 32/32
   Network greenbone-community-edition_default          Created
   Volume 'greenbone-community-edition_ospd_openvas_socket_vol'    Created
   Volume 'greenbone-community-edition_gpg_data_vol'      Created
   Volume 'greenbone-community-edition_gvmd_socket_vol'    Created...
   Volume 'greenbone-community-edition_notus_data_vol'     Created
   Volume 'greenbone-community-edition_redis_socket_vol'  Created..
   Volume 'greenbone-community-edition_vt_data_vol'        Created
   Volume 'greenbone-community-edition_gvmd_data_vol'      Created
   Volume 'greenbone-community-edition_cert_data_vol'     Created...
   Volume 'greenbone-community-edition_openvas_data_vol'   Created...
   Volume 'greenbone-community-edition_data_objects_vol'   Created...
   Volume 'greenbone-community-edition_sql_data_vol'       Created
   Volume 'greenbone-community-edition_sql_socket_vol'    Created...
   Volume 'greenbone-community-edition_openvas_log_data_vol' Created
   Volume 'greenbone-community-edition_scap_data_vol'     Created
Container greenbone-community-edition-vulnerability-tests-1 Started
Container greenbone-community-edition-gpg-data-1           Started
Container greenbone-community-edition-data-objects-1       Started...
Container greenbone-community-edition-cert-bind-data-1     Started...
Container greenbone-community-edition-redis-server-1       Started...
Container greenbone-community-edition-configure-openvas-1  St...
Container greenbone-community-edition-notus-data-1         Started
Container greenbone-community-edition-pg-gvm-1             Started
Container greenbone-community-edition-scap-data-1          Started
Container greenbone-community-edition-report-formats-1     Star...
Container greenbone-community-edition-dfn-cert-data-1     Start...
Container greenbone-community-edition-openvas-1            Started
Container greenbone-community-edition-ospd-openvas-1       Started...
Container greenbone-community-edition-openvasd-1           Started
Container greenbone-community-edition-gvmd-1              Started
Container greenbone-community-edition-gvmt-tools-1         Started
Container greenbone-community-edition-qsa-1               Started
```

3. Задаем пароль для админской учетной записи



```
astepanov@thinkpad:~/greenbone-community-container$ sudo docker-compose -f $DOWNLOAD_DIR/docker-compose.yml -p greenbone-community-edition exec -u gvmd gvmd gvmd --user=admin --new-password='123456'
[main] Message: 18:16:20.856:  Greenbone Vulnerability Manager version 25.1.3 (DB revision 258)
astepanov@thinkpad:~/greenbone-community-container$
```

4. Заходим в web-интерфейс развернутого приложения под учеткой **admin**



Шаг 2. Создать пользователя

Создаем пользователя Stepanov

The screenshot shows the 'Edit User Stepanov' dialog box open over the main application interface. The user 'Stepanov' is selected in the list on the left. The dialog contains the following fields:

- Login Name:** Stepanov
- Comment:** (empty)
- Authentication:** Password: Use existing Password New Password
- Roles:** Admin, Guest, Info, Monitor, Observer, User
- Groups:** (empty)
- Host Access:** Allow all and deny Deny all and allow

The background shows the main application interface with a sidebar containing 'Dashboards', 'Scans', 'Assets', 'Resilience', 'Security Information', 'Configuration', 'Administration' (selected), 'Users' (selected), 'Groups', 'Roles', 'Permissions', 'Performance', 'Trashcan', 'Feed Status', 'LDAP', 'RADIUS', 'Help'. On the right, there is a table of users with columns 'Host Access', 'Authentication Type', and 'Actions'.

Шаг 3. Настроить новый список для сканирования

Создаем новый список для сканирования согласно требованию в задании

The screenshot shows the Greenbone Security Assistant web interface in a Firefox browser. The URL is 127.0.0.1:9392/targets. The left sidebar is collapsed, showing the main navigation menu. The main content area displays a table for managing targets. A modal window titled "Hosts" is open, showing various configuration options for the target "localhost".

Name	Hosts	IPs	Port List	Credentials	Actions
localhost	192.168.1.0/24	253	All TCP and Nmap top 100 UDP		Edit Delete Copy Import

Hosts

- Included: 192.168.1.0/24
- Excluded: 192.168.1.100
- Maximum Number of Hosts: 253
- Allow simultaneous scanning via multiple IPs: Yes
- Reverse Lookup Only: No
- Reverse Lookup Unify: No
- Alive Test: Scan Config Default
- Port List: All TCP and Nmap top 100 UDP

Шаг 4. Создать новый таск на сканирование

Создаем новую задачу на сканирование под названием **HW Task**

The screenshot shows the Greenbone Security Assistant web interface in Firefox. The main title bar says 'Обзор Firefox' and 'Greenbone Security Assistant'. The address bar shows '127.0.0.1:9392/tasks'. The top right shows 'Вт, 1 апреля 22:54', '67%', and user 'admin @'. The left sidebar has a dark theme with various icons and a tree view of configuration sections like Dashboards, Scans, Tasks, Reports, etc. The 'Tasks' section is selected. The main content area is titled 'HW Task' and contains several tabs: 'Target', 'Alerts', 'Scanner', 'Assets', 'Schedule', and 'Scan'. Under 'Scanner', it shows 'Name: OpenVAS Default', 'Type: OpenVAS Scanner', 'Scan Config: Full and fast', 'Order for target hosts: sequential', 'Maximum concurrently executed NVTs per host: 4', 'Maximum concurrently scanned hosts: 20', 'Add to Assets: Yes', 'Apply Overrides: Yes', 'Min QoD: 70 %', and a 'Schedule' tab with 'Name: 2025 Weekly Schedule', 'Next: Tue, Apr 1, 2025 11:59 PM UTC', and 'Duration of last Scan: No scans yet'. At the bottom right of the content area is a 'Apply to page contents' button.

Внутри задачи создаем новое расписание по которому будем производить проверку на уязвимости в течение следующего года под названием **2025 Weekly Schedule**

The screenshot shows the 'Schedules' page with one entry: '2025 Weekly Schedule'. The table columns are 'Name', 'First Run', 'Next Run', 'Recurrence', 'Duration', and 'Actions'. The 'Actions' column includes edit and delete icons. The '2025 Weekly Schedule' row shows 'First Run: Tue, Apr 1, 2025 11:59 PM UTC', 'Next Run: Tue, Apr 1, 2025 11:59 PM UTC', 'Recurrence: Every week', 'Duration: 20 hours', and the 'Actions' row with edit and delete icons. At the bottom right of the table is a 'Apply to page contents' button.

Шаг 5. Создать алерт, который будет обращаться с помощью метода HTTP Get на <http://127.0.0.1:8000/alert> при появлении уязвимости с уровнем критичности >5.0

Создаем алерт "5+ vuln alert" согласно заданию и устанавливаем его в нашей задаче.

Edit Alert 5+ vuln alert

Comment
алерт, который будет обращаться с помощью метода HTTP Get на http://127.0.0.1:8000/alert n

Event

Task run status changed to

New NVTs

Ticket Received Assigned Ticket Changed Owned Ticket Changed

Condition

Always

Severity at least

Severity Level

Filter matches at least result(s) NVT(s)

Filter matches at least result(s) more than previous scan

Report Content

Compose

Delta Report

None

Previous completed report of the same task

Report with ID

Method

HTTP Get URL

Active

Yes No

Шаг 6. Сохранить таск

Страница с сохраненным таском

The screenshot shows the Greenbone Security Assistant web interface in a Firefox browser window. The URL is 127.0.0.1:9392/tasks. The left sidebar is dark-themed and includes sections for Dashboards, Scans, Tasks (selected), Reports, Results, Vulnerabilities, Notes, Overrides, Assets, Hosts, Operating Systems, TLS Certificates, Resilience, Security Information, Configuration, Targets, Port Lists, Credentials, Scan Configs, Alerts, Schedules, Report Configs, Report Formats, Scanners, Filters, Tags, and Administration (with sub-options for Users, Groups, and Roles). The main content area has a title 'Tasks 1 of 1'. It contains three visual summaries: 'Tasks by Severity Class (Total: 1)' (gray circle, N/A), 'Tasks with most High Results per Host' (empty chart), and 'Tasks by Status (Total: 1)' (green circle, 1 New). Below these are three tables: 'Tasks by Severity Class', 'Tasks with most High Results per Host', and 'Tasks by Status'. The 'Tasks by Status' table shows one row for 'HW Task' with status 'New'. At the bottom, there is a footer with copyright information: 'Copyright © 2009-2025 by Greenbone AG, www.greenbone.net'.