РАЗВЕРТЫВАНИЕ И НАСТРОЙКА СКАНЕРОВ УЯЗВИМОСТИ НА ОТЧЕСТВЕННЫХ ОС: CHKROOTKIT, LYNIS, RKHUNTER

Богданова Полина Александровна

Баев Ярослав Сергеевич

Студенты

Научный руководитель: Уймин Антон Григорьевич

Старший преподаватель

ФГАОУ ВО «РГУ НЕФТИ И ГАЗА ИМ. И. М. ГУБКИНА»

Аннотация:

В статье рассматривается процесс развертывания и установки сканеров уязвимости Chkrootkit, Lynis, Rkhunter на ОС таких как Alt Рабочая станция и Астра Linux. Рассматриваются специфические аспекты установки и конфигурации каждого из указанных инструментов. Приведены практические примеры использования сканеров в различных сценариях. Реализовано комбинирование и автоматизация сканеров Chkrootkit и Rkhunter.

Ключевые слова: Chkrootkit, Lynis, Rkhunter, сканер уязвимости, Alt Рабочая станция, Астра Linux.

DEPLOYMENT AND CONFIGURATION OF VULNERABILITY SCANNERS ON DOMESTIC OPERATING SYSTEMS: CHKROOTKIT, LYNIS, RKHUNTER

Bogdanova Polina Alexsandrovna

Baev Yaroslav Sergeevich

Students

Supervisor: Uymin Anton Grigorievich

Senior Lecturer

Gubkin Russian State University of Oil and Gas

Annotation:

The article discusses the deployment and installation process of vulnerability scanners Chkrootkit, Lynis, and Rkhunter on operating systems such as Alt Workstation and Astra Linux. It examines the specific aspects of the installation and configuration of each of the mentioned tools. Practical examples of using the scanners in various scenarios are provided. The combination and automation of Chkrootkit and Rkhunter scanners have been implemented.

Keywords: Chkrootkit, Lynis, Rkhunter, vulnerability scanner, Alt Workstation, Astra Linux.

ВВЕДЕНИЕ

С ростом информационных технологий и увеличением количества цифровых угроз вопрос обеспечения безопасности операционных систем стал приоритетным для государственных и частных компаний. Операционные системы типа Alt Рабочая станция разработаны с учетом национальных стандартов безопасности и применяются в различных отраслях - от государственного сектора до критической инфраструктуры. Несмотря на встроенные меры защиты, операционные системы остаются уязвимыми перед киберугрозами вроде руткитов, вредоносного ПО и атак на уязвимости систем. Поэтому важно регулярно проверять безопасность и осуществлять анализ на предмет уязвимостей для выявления и устранения потенциальных опасностей вовремя.

В данной статье рассматривается процесс развертывания и настройки инструментов для поиска уязвимостей и руткитов: Chkrootkit, Lynis и Rkhunter на платформах Alt Рабочая станция и Астра Linux. В работе будут рассмотрены возможности каждого инструмента, а также проведено

тестирование их эффективности и совместимости с данной операционной системой.

Chkrootkit (Check Rootkit) — это удобный для использования инструмент командной строки, разработанный для обнаружения руткитов и связанных угроз. Его основная функция заключается в анализе системных файлов и процессов на предмет наличия признаков руткитов.

Lynis — это продвинутый инструмент аудита безопасности, ориентированный на обнаружение уязвимостей и конфигурационных ошибок, обеспечивающий комплексный анализ системы. Он проводит аудит безопасности и выявляет уязвимые конфигурации, ошибки и рекомендации для их исправления.

Rootkit Hunter (Rkhunter) — это сканер, предназначенный для обнаружения руткитов, бекдоров и других уязвимостей. Rkhunter фокусируется на целостности системы, обнаружении подозрительных файлов и анализе изменений в системных конфигурациях.

Сводная таблица основных характеристик:

Таблица 1 — Сводная таблица характеристик

Название	**Разработчик**	**Цель разработки**	**Сфера применения**	**Лицензия**	**Использование в коммерческих продуктах**
Chkrootkit	Pangeia Informatica	Обнаружение руткитов в UNIX/Linux	Серверы, рабочие станции, ОС на основе UNIX	GPLv2	Не входит в коммерческие продукты напрямую
Lynis	CISOfy	Аудит безопасности и соответствия стандартам	Организации, аудиторы ИБ	GPLv3	Входит в состав некоторых продуктов аудита
RKHunter	Michael	Обнаружение	Домашние	GPLv2	Не

	Boelen	руткитов, троянов и вредоносных программ	системы, серверы, критически важные системы		используется в коммерческих продуктах напрямую

Для безопасного тестирования создадим виртуальную тестовую среду.

Установим Alt Рабочая станция и Астра Linux на виртуальной машине через VirtualBox с базовыми параметрами (2 CPU, 4 ГБ RAM, 20 ГБ диск) для проведения тестов.

Перейдем к описанию эксперимента.

ПОРЯДОК ЭКСПЕРИМЕНТА

1. Прежде всего необходимо подготовить и настроить сканеры.

1.1. Подготовка к установке сканеров.

Убедимся, что все зависимости и дополнительные утилиты для корректной работы сканеров установлены. Выполним следующие команды:

Для Alt Рабочая станция:

apt-get install gcc make glibc bash perl wget curl

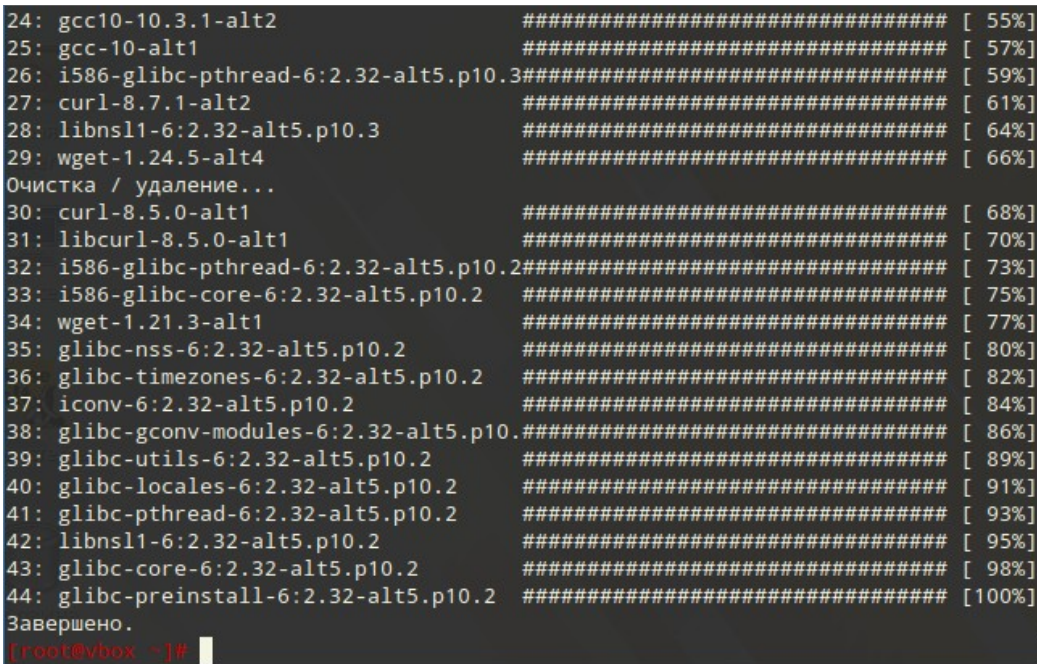

Рисунок 1 — Установка зависимостей и утилит. Alt Рабочая станция

Для Астра Linux:

sudo apt install gcc make glibc bash perl wget curl

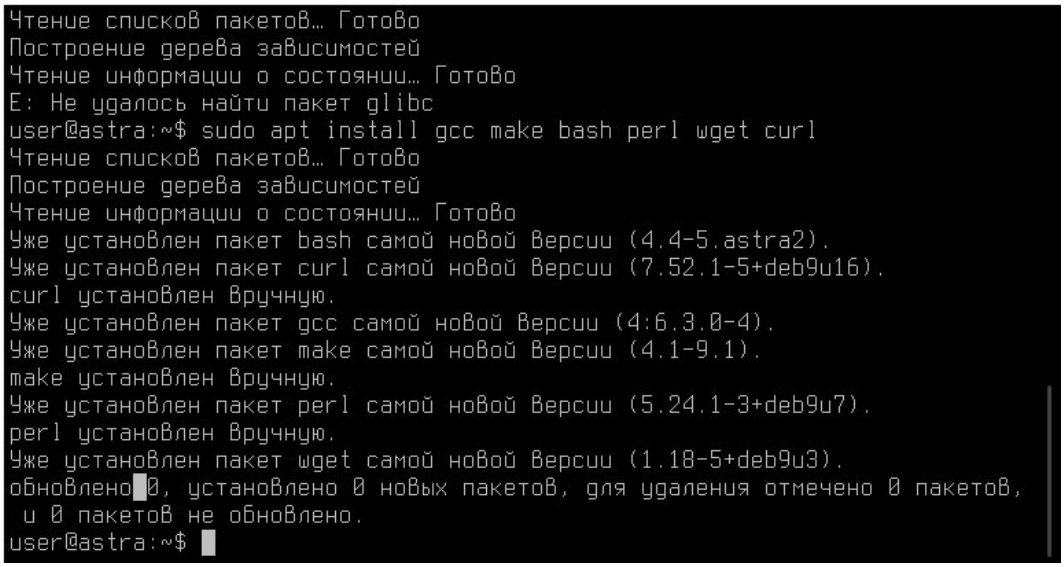

Рисунок 2 — Установка зависимостей и утилит. Астра Linux

1.2. Выполним обновление для установки последних версий пакетов, необходимых для стабильной работы сканеров:

Для Alt Рабочая станция:

apt-get update

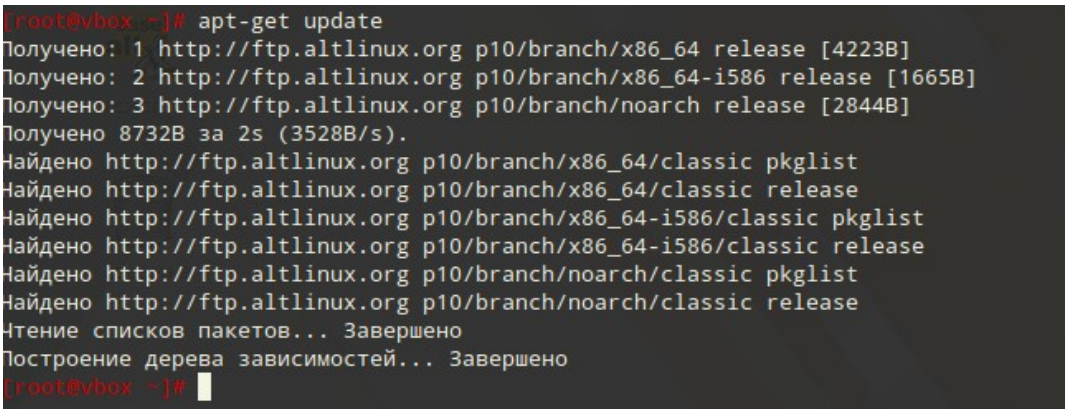

Рисунок 3 — Обновление операционной системы. Alt Рабочая станция

Для Астра Linux:

sudo apt update

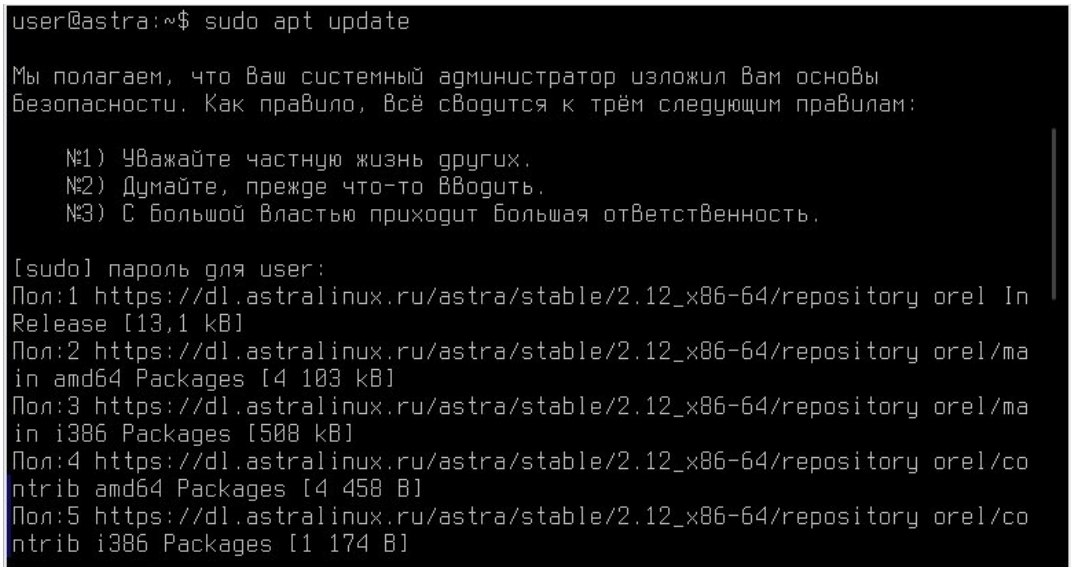

Рисунок 4 — Обновление операционной системы. Астра Linux

2. Установка и настройка сканеров.

2.1. Установка и настройка Chkrootkit. Выполним:

Для Alt Рабочая станция:

apt-get update

apt-get install chkrootkit -y

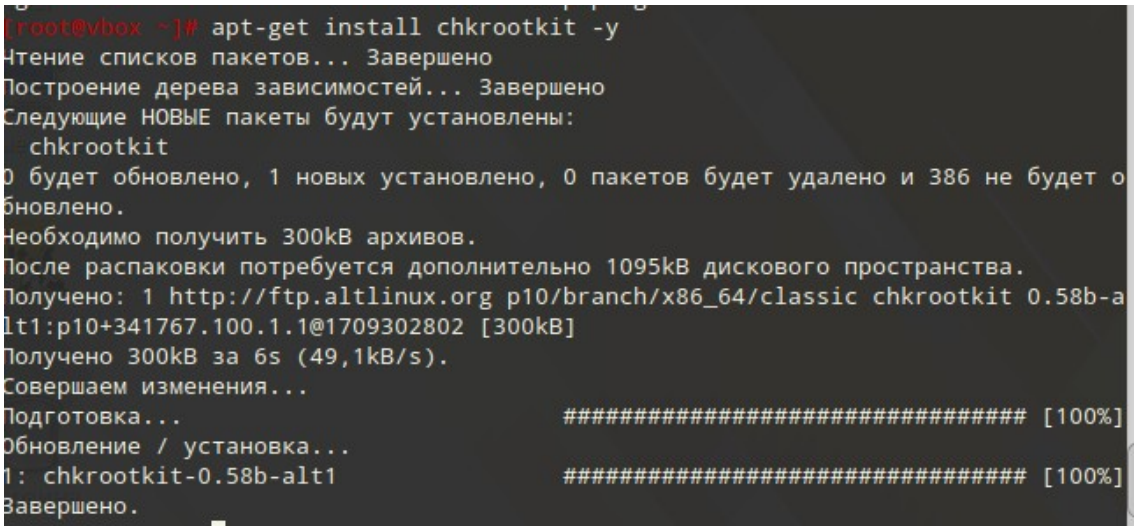

Рисунок 5 — Установка Chkrootkit. Alt Рабочая станция

Для Астра Linux:

sudo apt install chkrootkit

```
user@astra:~$ sudo apt install chkrootkit
Чтение списков пакетов… Готово
Построение дерева зависимостей
Чтение информации о состоянии… Готово
Уже установлен пакет chkrootkit самой новой версии (0.50-4+deb9u1).
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 0 пакетов,
 и 0 пакетов не обновлено.
```

Рисунок 6 — Установка Chkrootkit. Астра Linux

2.1.1. Запуск Chkrootkit и создание лог-файла. Для того чтобы автоматизированно сканировать систему и сохранять только подозрительные строки (например, содержащие INFECTED, warning или `suspicious`), используется следующий скрипт:

Для Alt Рабочая станция:

```
#!/bin/bash
LOG_FILE="/var/log/chkrootkit_filtered_$(date +%F_%H-%M-%S).txt"
chkrootkit | grep -iE "infected|suspicious|warning|detected" >> "$LOG_FILE"
if grep -q -iE "infected|suspicious|warning|detected" "$LOG_FILE"; then
    echo "Подозрительные элементы обнаружены. Лог записан в $LOG_FILE."
else
    echo "Система чиста. Угроз не обнаружено."
fi
```

```
  GNU nano 7.2                          /usr/local/bin/chkrootkit_audit.sh
#!/bin/bash
LOG_FILE="/var/log/chkrootkit_filtered_$(date +%F_%H-%M-%S).txt"
chkrootkit | grep -iE "infected|suspicious|warning|detected" >> "$LOG_FILE"
if grep -q -iE "infected|suspicious|warning|detected" "$LOG_FILE"; then
    echo "Подозрительные элементы обнаружены. Лог записан в $LOG_FILE."
else
    echo "Система чиста. Угроз не обнаружено."
fi
                                [ Прочитано 9 строк ]
^G Справка  ^O Записать  ^W Поиск  ^K Вырезать  ^T Выполнить  ^C Позиция  M-U Отмена
^X Выход    ^R ЧитФайл   ^\ Замена ^U Вставить  ^J Выровнять  ^/ К строке M-E Повтор
```

Рисунок 7 — Скрипт автоматического сканирования для Alt Рабочая станция

Для Астра Linux:

#!/bin/bash

LOG_FILE="/var/log/chkrootkit_filtered_$(date +%F_%H-%M-%S).txt"

sudo chkrootkit | grep -iE "infected|suspicious|warning|detected"| grep >> "$LOG_FILE"

if grep -q -iE "infected|suspicious|warning|detected" "$LOG_FILE"; then

echo "Подозрительные элементы обнаружены. Лог записан в $LOG_FILE."

else

echo "Система чиста. Угроз не обнаружено."

fi

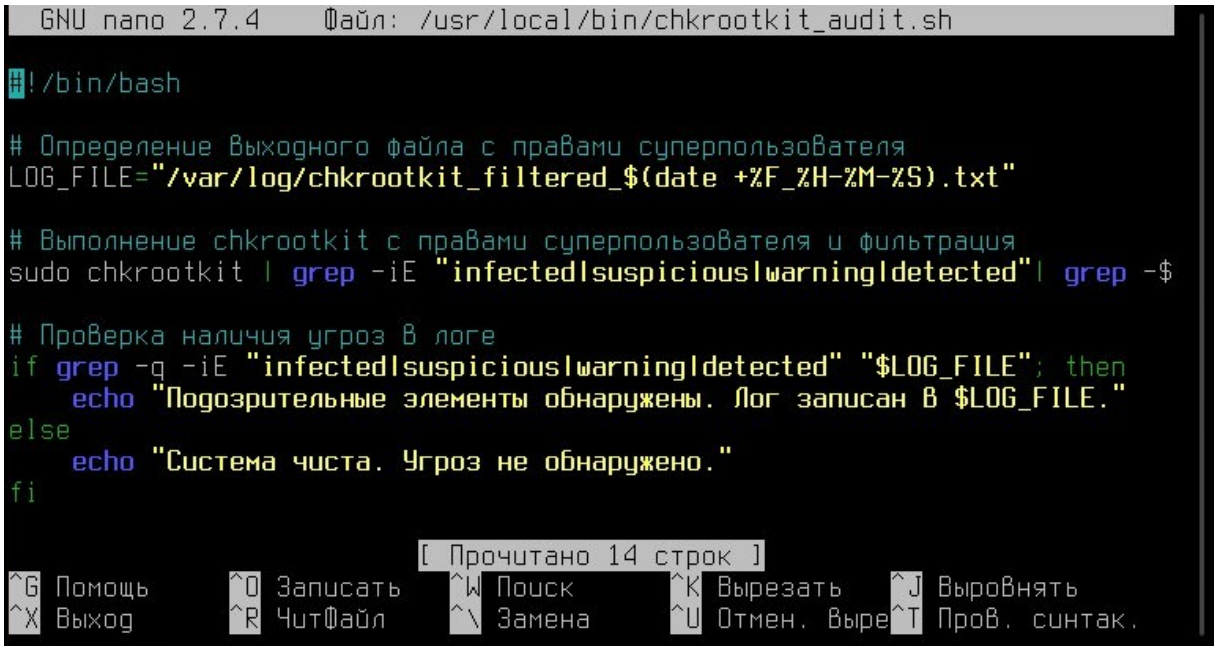

Рисунок 8 — Скрипт автоматического сканирования для Астра Linux

2.1.2. И сделаем скрипт исполняемым:

Для Alt Рабочая станция:

chmod +x /usr/local/bin/chkrootkit_audit.sh

Для Астра Linux:

sudo chmod +x /usr/local/bin/chkrootkit_audit.sh

2.2. Установка и настройка Lynis

Установим Lynis из стандартного репозитория для Альт Рабочая станция. Однако Лайнис для Астра Линукс нельзя так загрузить, поэтому

установим его с помощью официального репозитория Lynis на git.hub вручную:

Для Alt Рабочая станция:

apt-get install lynis

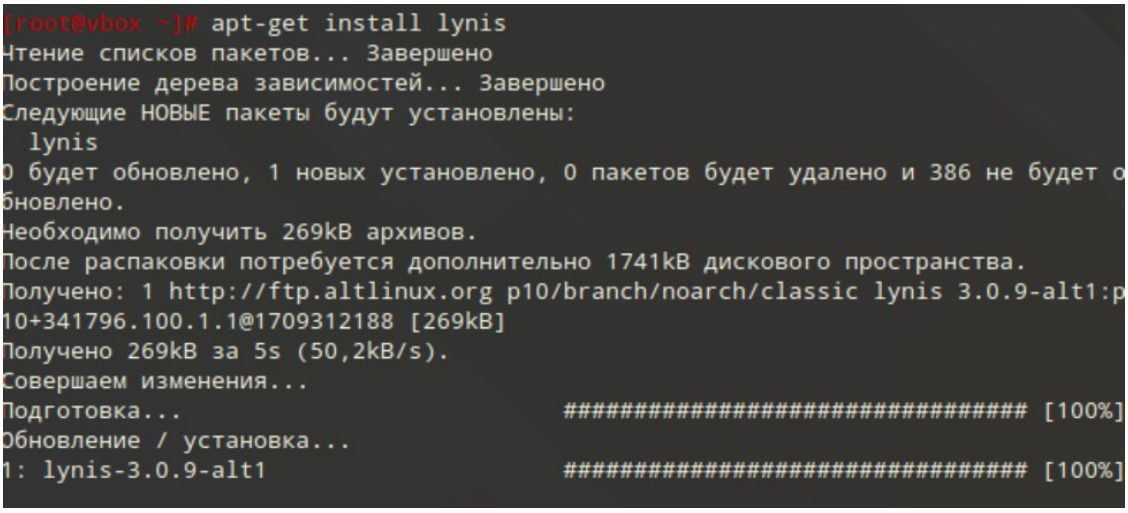

Рисунок 9 — Установка Lynis для Alt Рабочая станция

Для Астра Linux:

wget https://github.com/CISOfy/lynis/archive/refs/tags/3.0.0.tar.gz

tar -xvzf 3.0.0.tar.gz

Рисунок 10 — Установка Lynis для Астра Linux

Убедимся, что сканер успешно установился:

Для Alt Рабочая станция: *lynis show version*

Рисунок 11 — Активная версия Lynis для Alt Рабочая станция

Для Астра Linux: *./lynis --version*

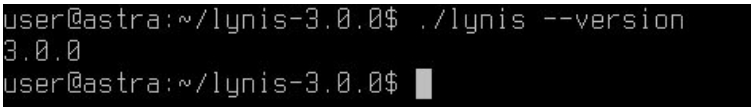

Рисунок 12 — Активная версия Lynis для Астра Linux

2.2.1. Настроим, чтобы сканер реагировал на слова «ПРЕДУПРЕЖДЕНИЕ|УЯЗВИМО|НЕБЕЗОПАСНО|ПРЕДЛОЖЕНИЕ»

Для того чтобы сохранять только подозрительные, создадим bash-скрипт, который будет автоматически запускать сканирование и запись по ключевым словам и выводить оценку безопасности:

Для Альт Рабочая станция:

#!/bin/bash

LOG_FILE=”/var/log/lynis_filtered_$(date +\%F_%H-%M-%S).txt”

echo “### Результаты аудита системы ###” > “$LOG_FILE”

lynis audit system | grep -iE “ПРЕДУПРЕЖДЕНИЕ|УЯЗВИМОСТЬ|НЕБЕЗОПАСНО|ПРЕДЛОЖЕНИЕ” >> “$LOG_FILE”

echo -e “\n### ###” >> “$LOG_FILE”

SYSTEM_SCORE=$(lynus show details | grep -I “hardening index” | awk ‘{print $NF}’)

echo “Оценка безопасности системы (Hardening Index): $SYSTEM_SCORE” >> “$LOG_FILE”

echo “Аудит завершен. Подозрительные операции записаны в $LOG_FILE”

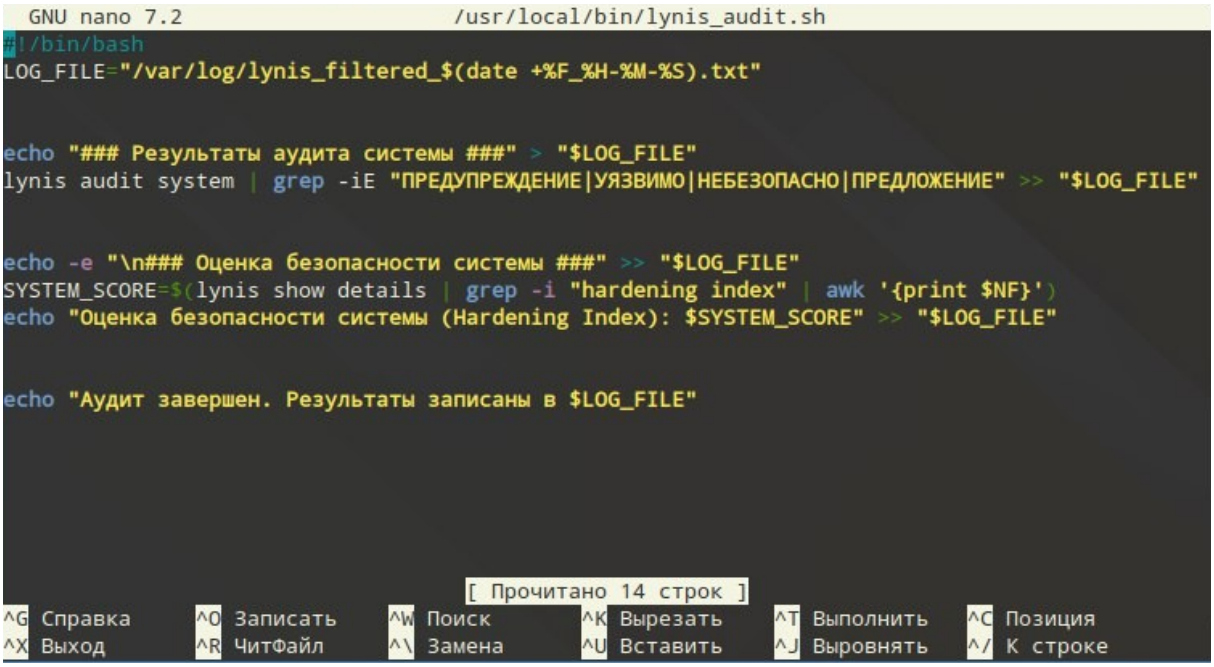

Рисунок 13 — bash-скрипт сканирования с фильтрацией для Alt Рабочая станция

Необходимо сделать файл исполняемым:

chmod +x /usr/local/bin/lynus_audit.sh

Для Астра Linux:

#!/bin/bash

LOG_FILE1="/var/log/lynis_filtered_$(date +%F_%H-%M_%S).txt"

./lynis audit system --quiet | grep -iE "warning|vulnerable|risk|unsafe" | sudo tee "$LOG_FILE1" > dev/null

echo "Отчет Lynis сохранен в $LOG_FILE1"

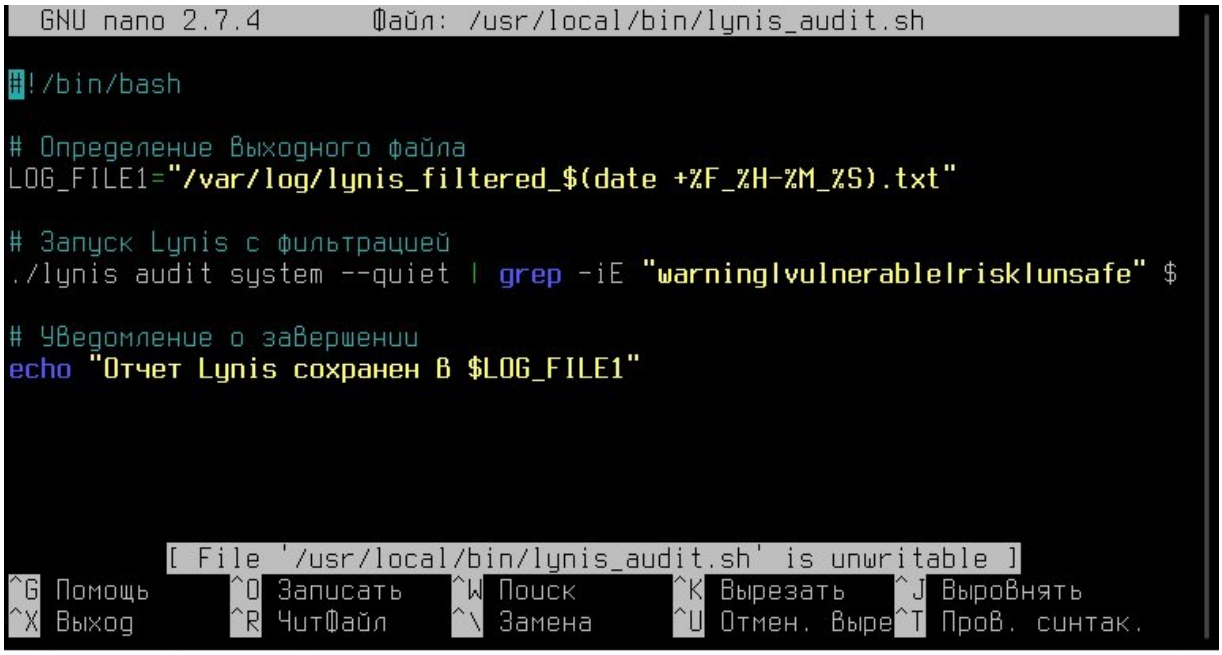

Рисунок 14 — bash-скрипт сканирования с фильтрацией для Астра Linux

2.3. Установка и настройка Rkhunter.

Для Альт Рабочая станция:

Установим сканер из стандартного репозитория и настроим скрипт для выполнения сканирования и вывода в лог-файл только предупреждающих строк:

nano /usr/local/bin/rkhunter_audit.sh

```
#!/bin/bash
LOG_FILE2="/var/log/rkhunter_filtered_$(date +%F_%H-%M-%S).txt"
rkhunter --check --skip-keypress | grep -iE "Invalid|Warning|Vulnerable|Suspicious|Alert" > "$LOG_FILE2"
if grep -q -iE " Invalid|Warning|Vulnerable|Suspicious|Alert" " "$LOG_FILE2"; then
   echo "Подозрительные элементы обнаружены. Лог записан в $LOG_FILE2."
else
   echo "Система чиста. Угроз не обнаружено."
fi
```

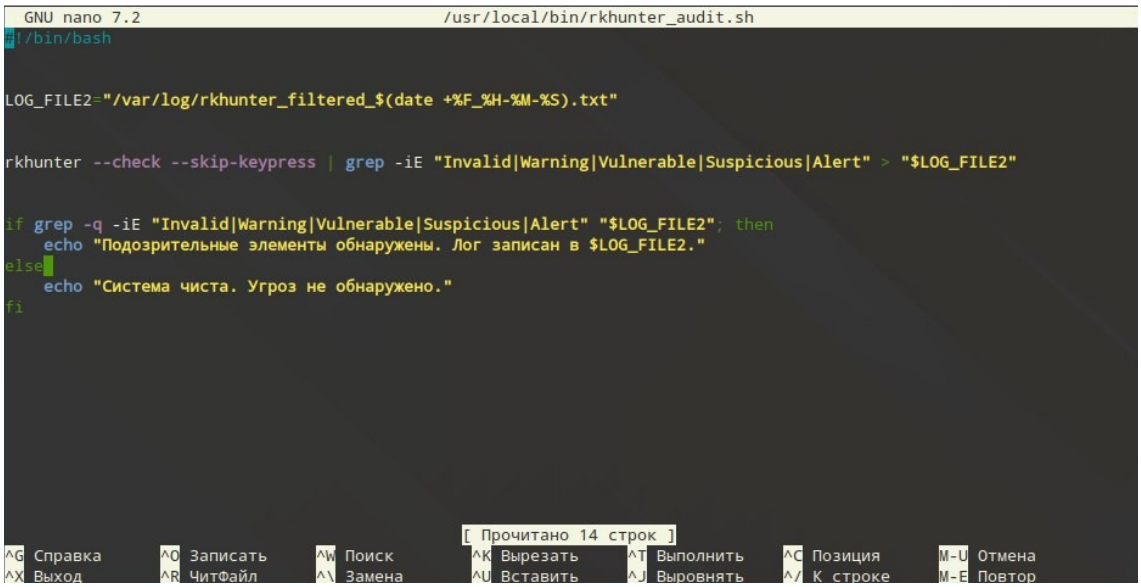

Рисунок 15 — Скрипт фильтрации Rkhunter 1

И протестируем, запустив сканирование:

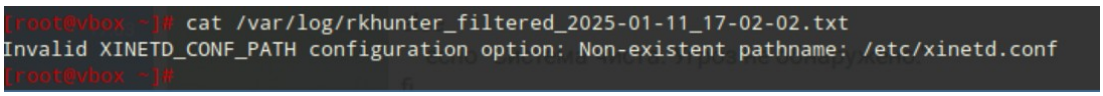

Рисунок 16 — Обнаружение некорректной работы

Для Астра Linux:

Для Астра же требуется установка из другого источника SourceForce, так как в стандартном репозитории его нет. Для этого требуется написать следующие команды:

sudo wget

https://sourceforge.net/projects/rkhunter/files/rkhunter/1.4.6/rkhunter-1.4.6.tar.gz

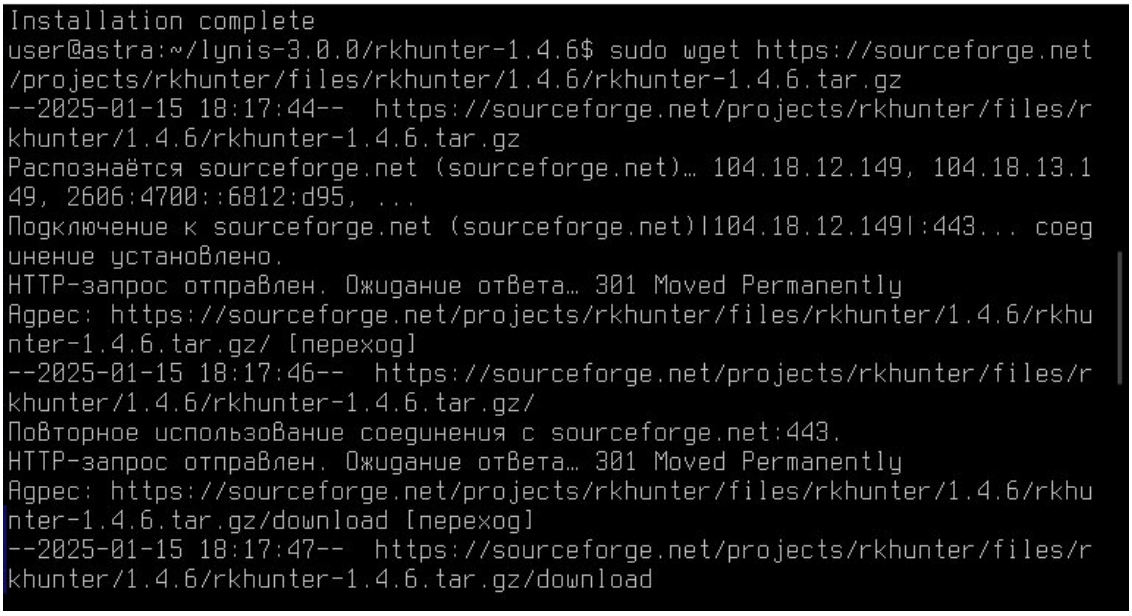

Рисунок 17 — Установка с помощью стороннего ресурса

И распаковать архив. После чего необходимо запустить установщик:

sudo ./installer.sh --layout /usr/local --install

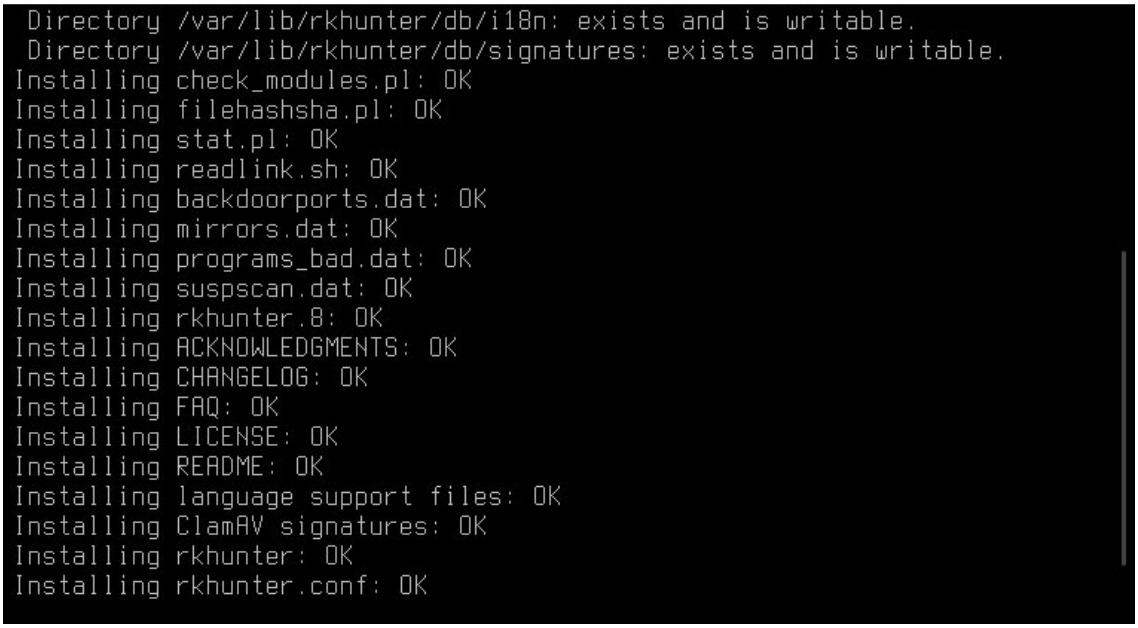

Рисунок 18 — Запуск установщика

Настроим фильтрацию строк:

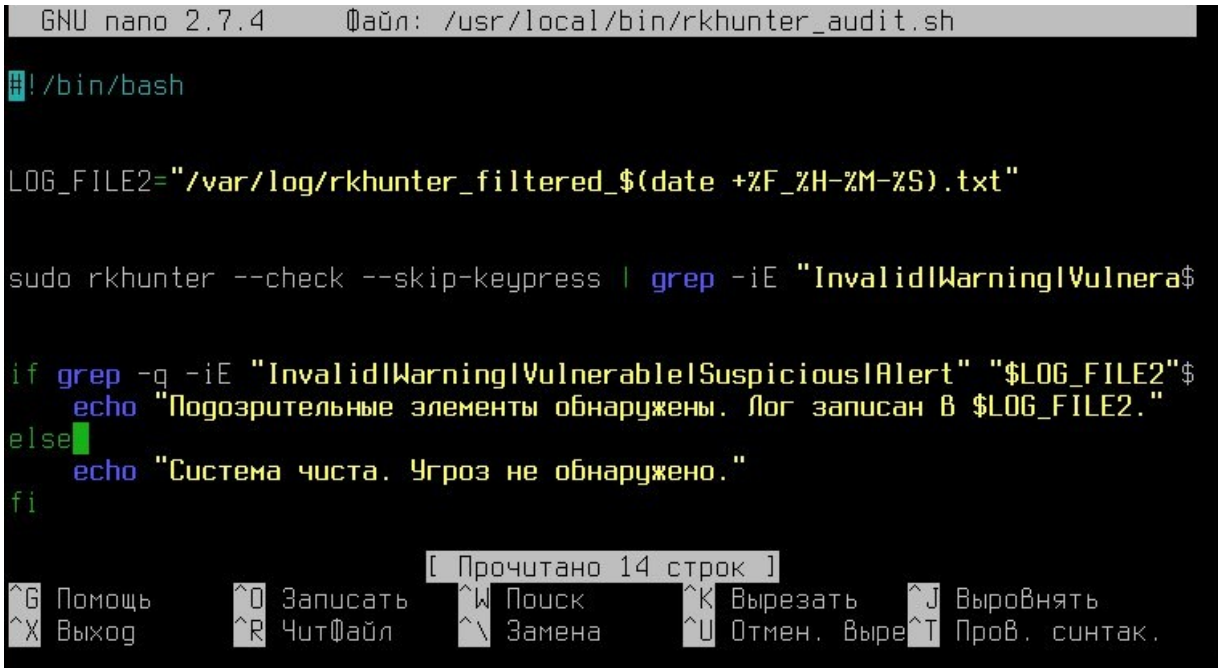

Рисунок 19 — Скрипт фильтрации Rkhunter 2

3. Комбинирование и автоматизация сканеров.

Создадим скрипт, который будет выполнять поочередное выполнение сканеров и записи в единый лог-отчет:

#!/bin/bash

LOG_FILE="/var/log/chkrootkit_filtered_$(date +%F_%H-%M-%S).txt"

LOG_FILE2="/var/log/rkhunter_filtered_$(date +%F_%H-%M-%S).txt"

COMBINED_LOG="/var/log/combined_scan_$(date +%F_%H-%M-%S).txt"

DATE=$(date '+%Y-%m-%d %H:%M:%S')

chkrootkit | grep -iE "infected|suspicious|warning|detected" | grep -ivE "not infected|nothing detected|Searching fo>

rkhunter --check --skip-keypress | grep -iE "Invalid|Warning|Vulnerable|Suspicious|Alert" > "$LOG_FILE2"

echo "=== Объединенный отчет о безопасности: $DATE ===" > "$COMBINED_LOG"

echo "=== Результаты Chkrootkit ===" >> "$COMBINED_LOG"

cat "$LOG_FILE" >> "$COMBINED_LOG"

echo "" >> "$COMBINED_LOG"

echo "=== Результаты RKHunter ===" >> "$COMBINED_LOG"

cat "$LOG_FILE2" >> "$COMBINED_LOG"

echo "Выполнено совместное сканирование RKhunter и Chkrootkit. Просмотреть общий лог можно в $COMBINED_LOG"

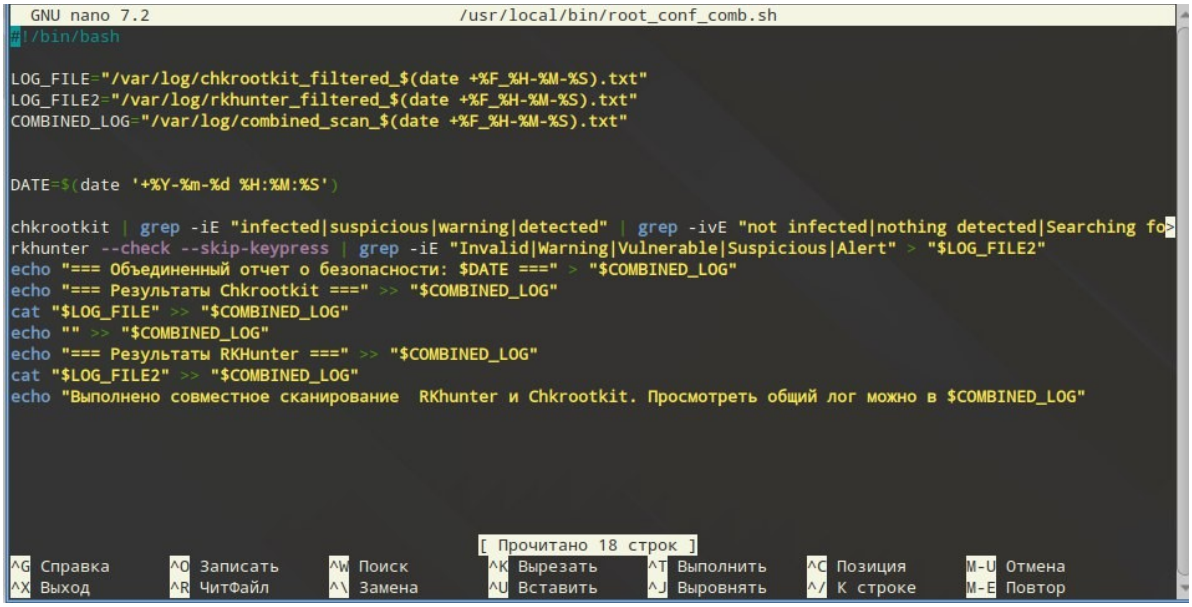

Рисунок 20 — Скрипт объединения логов сканеров

Таким образом после того, как мы сделали файл выполняемым, можно проверить итоговый вывод:

root_conf_comb.sh

cat /var/log/combined_scan_2025-01-11_18-19-12.txt

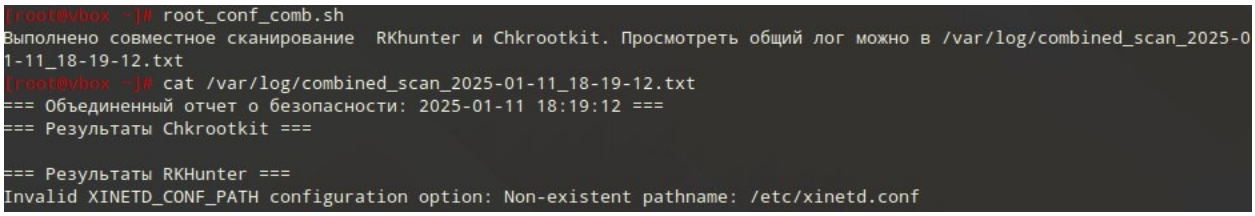

Рисунок 21 — Отчет сканирования

Объединенный сканер успешно отработал, поэтому можно создать Cron-задание для ежедневной исполняемой проверки операционной системы:

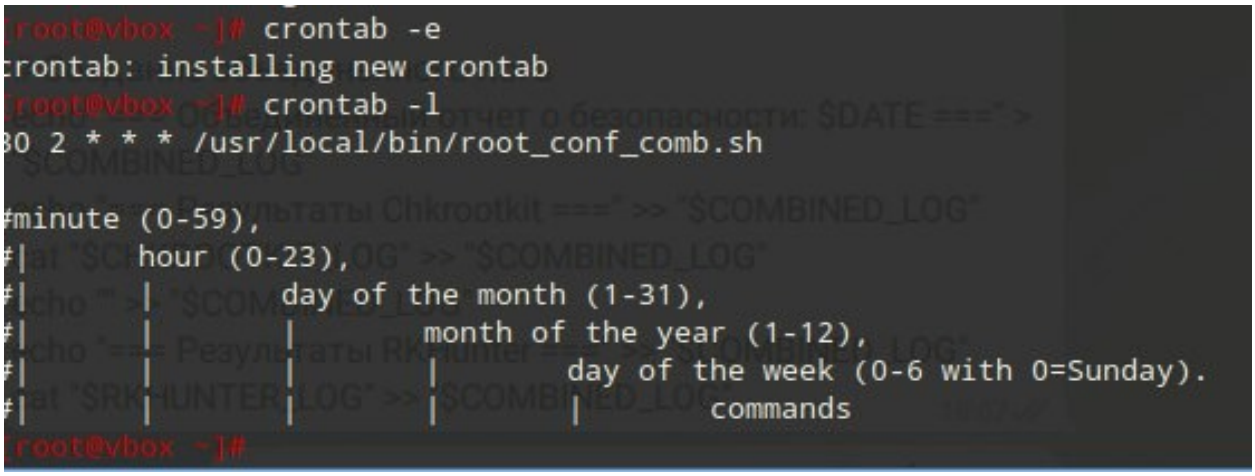

Рисунок 22 — Cron автоматизация

Таким образом, мы автоматизировали работу системы мониторинга и уплотнили безопасность ОС, сканирование будет выполняться ежедневно в 2:30.

Для полного охвата безопасности операционной системы установим Cron задание на выполнение полного еженедельного сканирования с помощью Lynis:

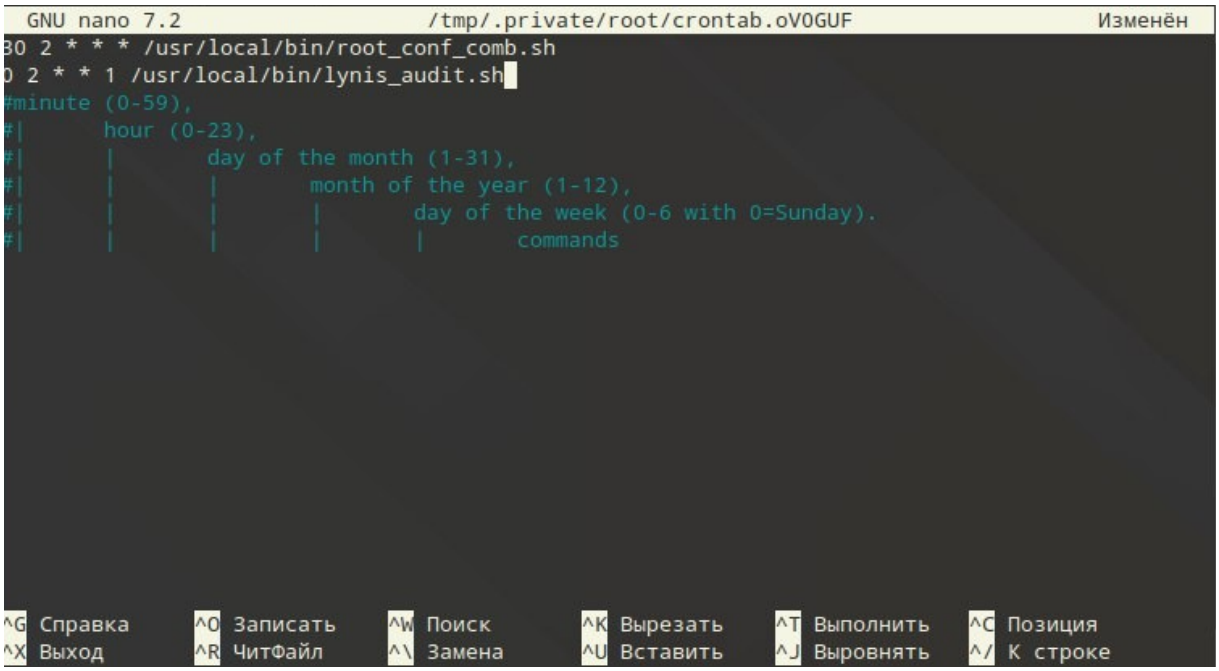

Рисунок 23 — Cron автоматизация Lynis

4. Экспериментальная часть.

Создадим файл с содержимым теста EICAR, файл не представляет угрозы, он нужен для проверки работоспособности сканеров и антивирусов

echo 'X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H' > /tmp/eicar.com*

Убедимся, что файл создан:

cat /tmp/eicar.com

Запустим скрипт:

root_conf_comb.sh

RKHunter и Chkrootkit должны распознать eicar.com как потенциальную угрозу.

Однако сканеры вывели пустой результат, это ожидаемое поведение, поскольку эти утилиты не предназначены для обнаружения антивирусных тестовых файлов. Их основная задача — обнаружение руткитов, скрытых процессов, изменённых системных файлов и следов компрометации.

В таком случае попробуем создать скрытые процессы и имитировать руткит:

nohup sleep 9999 &

nohup bash -c "while :; do echo 'hidden'; done" &

И снова запустим скрипт объединения сканеров:

root_conf_comb.sh

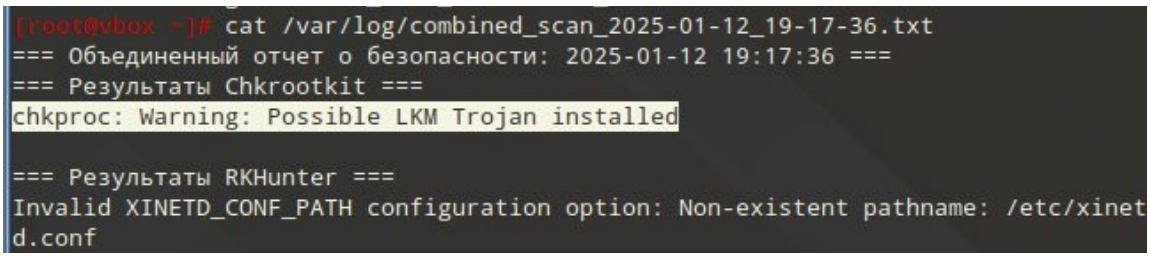

Рисунок 24 — Обнаружение уязвимости

Это предупреждение говорит о том, что сканер Chkrootkit обнаружил аномалию, связанную с процессами, которая может быть признаком LKM-трояна.

Также из git-hub установим руткит, который часто применяется в лабораторных условиях для тестирования системы безопасности – adore-ng. Эксперимент выполняется на изолированной машине в Virtual Box и используется исключительно в научно-ознакомительных целях, что не противоречит законодательству РФ. С помощью него проведем контрольную проверку:

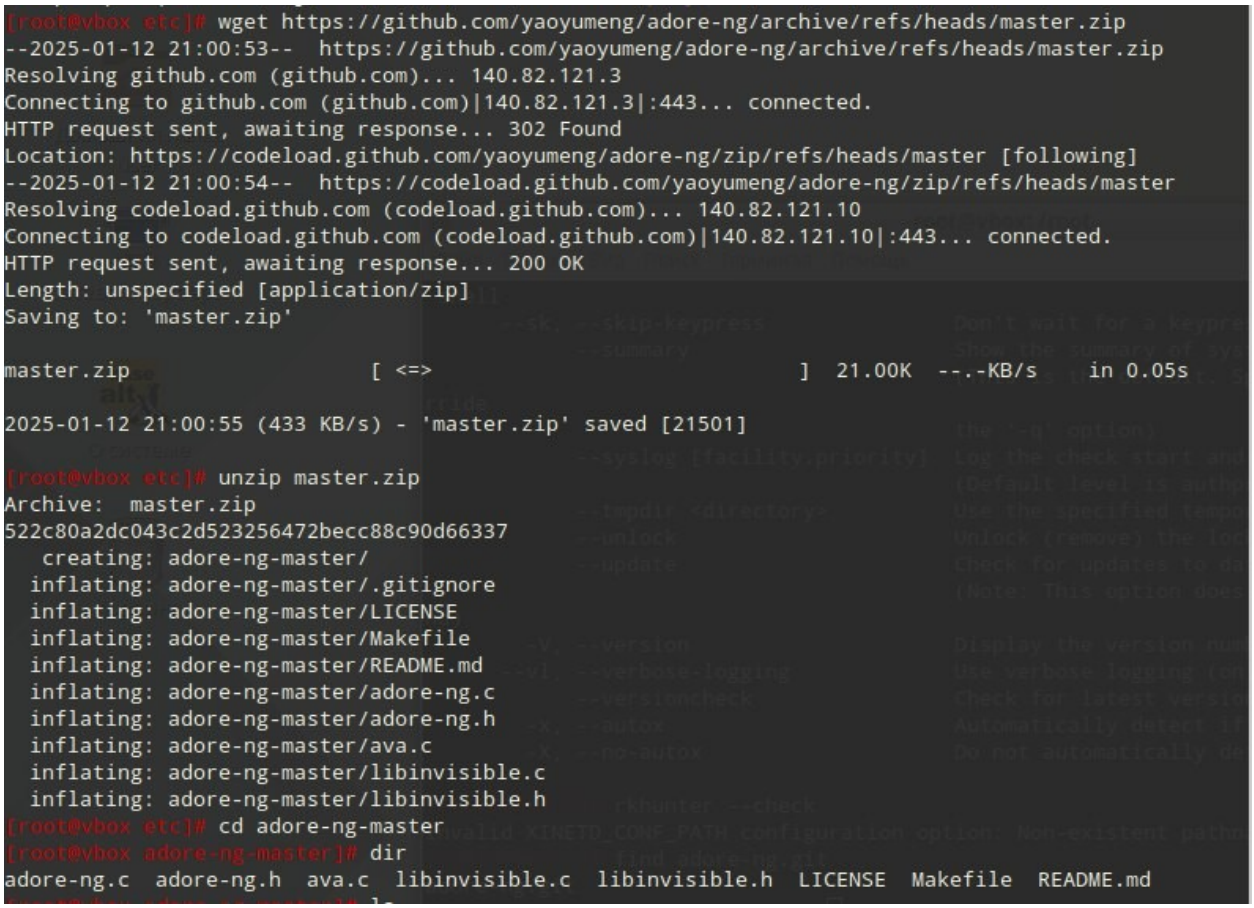

Рисунок 25 — Установка руткита

Руткит успешно установлен, поэтому запускаем скрипт:

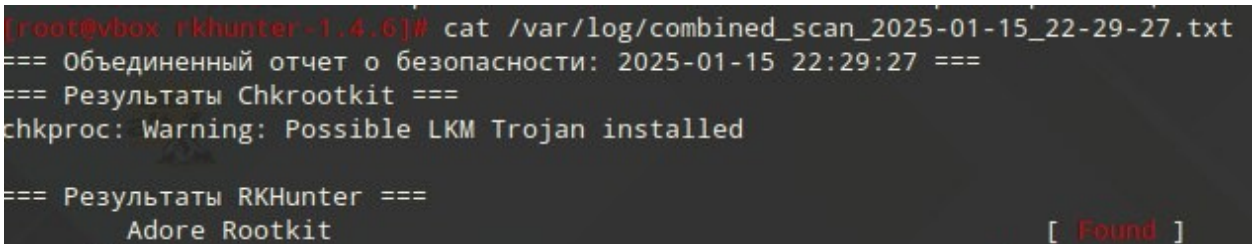

Рисунок 26 — Обнаружение руткита

Сканеры успешно обнаружили руткит и обнаружил аномалию.

4.1 Тестирование Lynis

Сделаем проверку многоступенчатой, чтобы оценить многосторонность сканера и его комплексную эффективность.

4.1.1 Проверка прав на файлы и каталоги

Цель: убедиться, что Lynis отслеживает неправильные права на чувствительные файлы

chmod 777 /etc/sudoers

Проверка прав на домашние директории:

chmod 777 /home/user

Lynis обнаружил неправильные права на чувствительные файлы, такие как /etc/sudoers:

Было обнаружено предупреждение о правах на файл /etc/sudoers.

Также были отмечены и домашние директории, которые могут привести к угрозам безопасности.

4.1.2 Проверка sysctl и параметров ядра

Цель: проверить, как Lynis реагирует на изменения в параметрах ядра через sysctl.

Изменение параметров ядра:

sysctl -w kernel.randomize_va_space=0

sysctl -w net.ipv4.conf.all.rp_filter=0

echo "kernel.randomize_va_space=0" >> /etc/sysctl.conf

echo "net.ipv4.conf.all.rp_filter=0" >> /etc/sysctl.conf

Lynis обнаружил небезопасные параметры ядра:

Параметры kernel.randomize_va_space=0 и net.ipv4.conf.all.rp_filter=0, были помечены как небезопасные, так как эти изменения могут ослабить защиту от атак, связанных с утечками памяти или сетевыми атаками.

4.1.3 Проверка сетевых настроек

Цель: проверить, как Lynis оценивает настройки сети и брандмауэра.

Отключение брандмауэра:

systemctl stop firewalld

systemctl disable firewalld

Изменение конфигурации hosts.allow и hosts.deny:

echo "ALL: ALL" > /etc/hosts.allow

echo "ALL: ALL" > /etc/hosts.deny

Так как был отключен брандмауэр firewalld и неправильно настроены файлы /etc/hosts.allow и /etc/hosts.deny, Lynis корректно отметил это как проблему. Установка значений "ALL: ALL" в эти файлы — прямая угроза безопасности, так как открывает доступ ко всем системам без ограничений.

4.1.4 Проверка учетных записей и паролей

Цель: проверить, как Lynis реагирует на слабые пароли и неправильные настройки для пользователей.

Создание или изменение учетных записей со слабыми паролями:

useradd testuser

echo "123" | passwd --stdin testuser

Изменение параметров в /etc/login.defs:

echo "PASS_MIN_LEN 3" >> /etc/login.defs

Были созданы пользователи с слабыми паролями - testuser с паролем "123", и были внесены изменения в файл /etc/login.defs, позволяющие использовать слабые пароли - PASS_MIN_LEN 3, и Lynis сообщил об этих слабых конфигурациях.

Рисунок 27 — Обнаружение уязвимости Lynis

Таким образом, Lynis также успешно справился с тестом.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Учебные, научные и иные публикации

1. Образцов Е. С. Мониторинг как средство защиты информации // Перспективы развития информационных технологий. – 2012. – №9. – С. 129-131.

Эмпирические материалы

2. Вызовы информационной безопасности для ТЭК в условиях существующих ограничений и трендов импортозамещения. – URL: https://www.novostiitkanala.ru.

3. Какие киберугрозы ждут нефтегазовых гигантов в 2024 году. – URL: https://companies.rbc.ru.

4. Крупнейшие утечки данных в 2024: как хакеры отнимают у нас приватность – URL: https://gerwin.io.

5. Утечки персональных данных в России: статистика первого полугодия 2024 – URL: https://www.cibit.ru.

6. Innostage: почти в 80% инцидентов ИБ напрямую виноват человеческий фактор. – URL: https://companies.rbc.ru.

7. Positive Technologies: число атак на промышленность выросло на 53%. – URL: https://www.ptsecurity.com.