

## Лабораторная работа №8.

### Средства обнаружения угроз в Linux

Цель работы: Изучение инструментов обнаружения угроз и мониторинга безопасности в системах Linux.

#### 1. OpenSCAP

Отслеживание изменений в ОС и выхода обновлений является сложной задачей, требующей автоматизации. Одним из популярных методов автоматизации аудита безопасности является протокол *SCAP* (*Security Content Automation Protocol*), разработанный *NIST* (*National Institute of Standards and Technology*) в 2009 году. *SCAP* обеспечивает стандартизированное управление уязвимостями и оценку соответствия политиками.

Компонентом *SCAP* является язык *OVAL* (*Open Vulnerability and Assessment Language*), который стандартизует процесс оценки безопасности, включая представление информации о конфигурациях, анализ состояния системы и отчет о результатах.

Пример реализации *SCAP* — проект *OpenSCAP*, который позволяет проверять параметры конфиденциальности системы и выявлять признаки компрометации с использованием стандартных правил.

Для настройки или сканирования уязвимостей локальной системы необходим сам инструмент (например, *oscap* или *SCAP Workbench*) и содержимое *SCAP* (включая потоки данных *SCAP*, *XCCDF*, *OVAL* и др.).

*OpenSCAP* можно собрать вручную из исходного кода, предоставляемого разработчиком продукта либо установить существующую сборку для ОС “Альт Рабочая станция” из репозитория.

Установим пакеты *openscap-scanner* и *openscap-utils*:

```
# apt-get update && apt-get install openscap-scanner openscap-utils
```

Выведем версию *OpenSCAP*:

```
# oscap --version
```

или

```
# oscap -V
```

```
[root@alt-1 ~]# oscap --version
OpenSCAP command line tool (oscap) 1.3.10
Copyright 2009--2023 Red Hat Inc., Durham, North Carolina.

==== Supported specifications ====
SCAP Version: 1.3
XCCDF Version: 1.2
OVAL Version: 5.11.1
CPE Version: 2.3
CVSS Version: 2.0
CVE Version: 2.0
Asset Identification Version: 1.1
Asset Reporting Format Version: 1.1
CVRP Version: 1.1

==== Capabilities added by auto-loaded plugins ====
No plugins have been auto-loaded...

==== Paths ====
Schema files: /usr/share/openscap/schemas
Default CPE files: /usr/share/openscap/cpe
```

Версия инструмента *OpenSCAP* 1.3.10.

Команда *oscap* проверяет систему на соответствие определенным стандартам.

Синтаксис команды *oscap*:

# oscap [options] module operation [operation\_options\_and\_arguments]

Разберем несколько типов модулей:

Название модуля	Описание
CPE (Common Platform Enumeration)	Определяет и описывает программные платформы и ОС с помощью унифицированных идентификаторов.
CVE (Common Vulnerabilities and Exposures)	Предоставляет уникальные идентификаторы известных уязвимостей в ПО.
CVSS (Common Vulnerability Scoring System)	Оценивает уязвимости по набору метрик.
DS (Data Stream)	Определяет формат для передачи данных о безопасности между системами.
OVAL (Open Vulnerability and Assessment Language)	Оценивает уязвимости по каждому определению в файле.
INFO	Определяет тип файла и выводит информацию о нем.
XCCDF	Определяет формат для описания контрольных списков конфигурации.

Значения для *operation* зависят от типа модуля. Разберем часто используемые операции с модулями *OVAL* и *XCCDF*:

Название операции	Модуль OVAL	Модуль XCCDF
eval	Проверяет систему, оценивает каждое определение в файле и выводит результаты в стандартный вывод.	Проверяет систему на соответствие каждому правилу в файле и выводит результаты в стандартный вывод.
generate	<i>generate report</i> преобразует указанный файл в отчет в формате <i>HTML</i> .	<i>generate guide</i> выводит полное руководство по безопасности для указанного профиля.
validate	Проверяет файл <i>OVAL</i> или <i>XCCDF</i> на соответствие XML-схеме.	

Основная цель *OpenSCAP* заключается в настройке и сканировании уязвимостей локальной системы. *OpenSCAP* может оценивать исходные потоки данных *SCAP*, эталонные тесты *XCCDF* и определения *OVAL* и генерировать соответствующие результаты. Содержимое *SCAP* может предоставляться либо в одном файле (как исходный поток данных *SCAP*), либо в виде нескольких отдельных XML-файлов.

Все XML-файлы расположены в каталоге */usr/share/openscap*.

Выведем информацию о файле *SCAP openscap-cpe-oval.xml*:

```
# oscap info /usr/share/openscap/cpe/openscap-cpe-oval.xml
```

```
[root@alt-1 ~]# oscap info /usr/share/openscap/cpe/openscap-cpe-oval.xml
Document type: OVAL Definitions
OVAL version: 5.10.1
Generated: 2012-11-22T15:00:00+01:00
Imported: 2024-04-16T22:15:35
```

*Document type* указывает, что документ является определением *OVAL*.

*OVAL version* указывает на версию спецификации *OVAL* 5.10.1, которая используется в данном документе.

*Generated* указывает на дату и время, когда был сгенерирован этот документ. В данном случае 22 ноября 2012 года.

*Imported* указывает на то, что документ был загружен или обновлен 16 апреля 2024 года.

*OpenSCAP* обрабатывает файл определений *OVAL* во время оценки определений *OVAL*. Инструмент собирает системную информацию, оценивает ее и создает файл результатов *OVAL*. Также результат оценки каждого определений *OVAL* выводится в стандартный поток вывода.

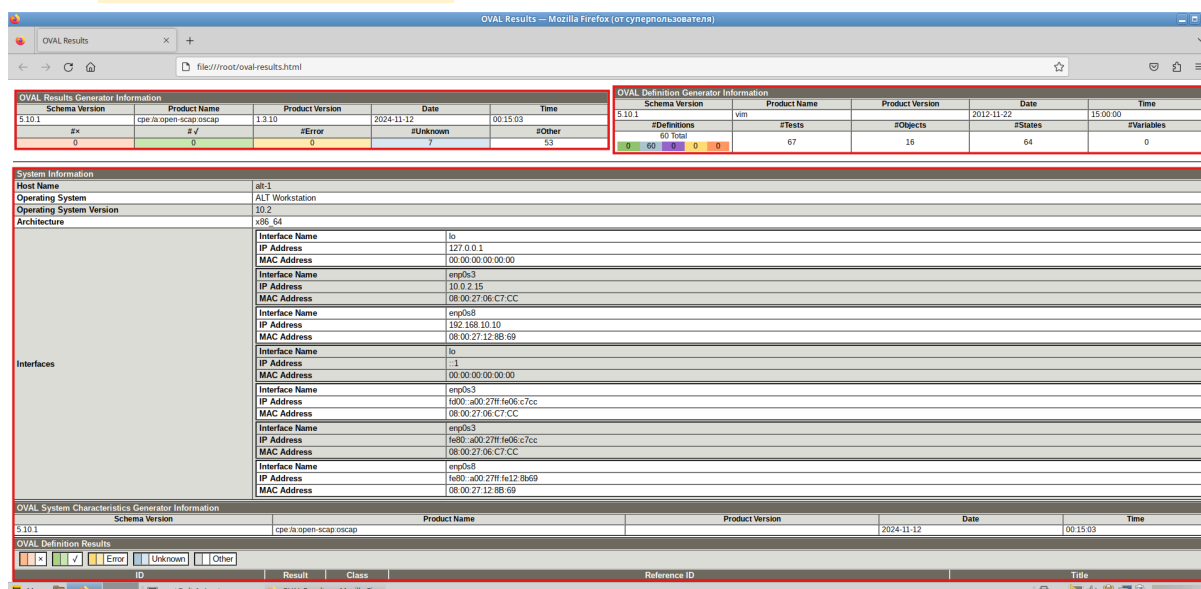
Для оценки определений OVAL выполним следующую команду:

```
# oscap oval eval --results oval-results.xml --report oval-results.html /usr/share/openscap/cpe/openscap-cpe-oval.xml
```

Вместе с выводом результатов в стандартный поток вывода генерируются удобочитаемый отчёт *oval-results.html* и машиночитаемый отчет *oval-results.xml*.

Откроем файл *oval-results.html*:

```
# firefox oval-results.html
```



OVAL Results Generator Information						OVAL Definition Generator Information					
Schema Version	Product Name	Product Version	Date	Time		Schema Version	Product Name	Product Version	Date	Time	
5.10.1	cpe:/a:open-scap:oscap	1.3.10	2024-11-12	00:15:03		5.10.1	vim		2012-11-22	15:00:00	
#x	#v	#Error	#Unknown	#Other		#Definitions	#Tests	#Objects	#States	#Variables	
0	0	0	7	53		60	67	16	64	0	

System Information	
Host Name	alt-1
Operating System	ALT Workstation
Operating System Version	10.2
Architecture	x86_64

Interfaces	
Interface Name	lo
IP Address	127.0.0.1
MAC Address	00:00:00:00:00:00
Interface Name	ens3
IP Address	10.0.2.15
MAC Address	08:00:27:06:C7:CC
Interface Name	ens3
IP Address	192.168.10.10
MAC Address	08:00:27:12:8B:69
Interface Name	lo
IP Address	1
MAC Address	00:00:00:00:00:00
Interface Name	ens3
IP Address	100:a0:27ff:fe06:c7cc
MAC Address	08:00:27:06:C7:CC
Interface Name	ens3
IP Address	100:a0:27ff:fe06:c7cc
MAC Address	08:00:27:06:C7:CC
Interface Name	ens3
IP Address	100:a0:27ff:fe12:8b69
MAC Address	08:00:27:12:8B:69

OVAL System Characteristics Generator Information					
Schema Version	Product Name	Product Version	Date	Time	
5.10.1	cpe:/a:open-scap:oscap		2024-11-12	00:15:03	

OVAL Definition Results					
ID	Result	Class	Reference ID	Title	

Условно отчет можно разделить на три основные части:

1. Информация о генераторе результатов OVAL содержит версию *oscap*, использованную при сканировании, дату и время операции и итоговые сведения об обнаруженных проблемах.

2. Информация о генераторе определений OVAL содержит основную информацию о файле определений OVAL — название, дату и время его выпуска, количество определений и тестов.

3. Результаты сканирования в первую очередь содержат информацию об исследуемом хосте: его имя, название ОС, версию и архитектуру. Также дана дополнительная расширенная информация об обнаруженных хостах ОС. Самая важная часть отчета — результаты проверки актуальности установленных пакетов ОС.

Строки отчета имеют цветовое выделение согласно статусу обнаруженной проблемы: зеленые строки отмечают пакеты, которые не установлены в системе или имеют актуальную версию, оранжевым отмечены пакеты, требующие обновления.

Таблица описания имеет следующие столбцы:

*ID* — уникальный идентификатор описания, присваиваемый производителем ОС;

*Result* — результат выполнения проверки: *false* — если проблем не обнаружено, *true* — требуется исправление недостатка;

*Class* — в данном случае всегда принимает значение *vulnerability* (уязвимость);

*Reference ID* — ссылка на бюллетень производителя, где хранится дополнительная информация об уязвимости;

*Title* — заголовок названия бюллетеня, позволяющий понять, о какой уязвимости идёт речь.

Файл *oval-results.xml* содержит те же сведения, что и html-файл отчета, но более удобен для использования в других системах анализа и сканирования уязвимостей.

Выполним сканирование удаленного хоста *alt-2*.

Имя хоста	IP-адрес
alt-1 (сервер)	192.168.10.10/24
alt-2 (клиент)	192.168.10.20/24

**Примечание!** Локальная сеть между машинами настроена по умолчанию.

Подключимся к хосту *alt-2* и установим сканер *OpenSCAP*:

```
# ssh user@192.168.10.20
```

```
$ su -
```

```
# apt-get update && apt-get install openscap-scanner
```

Отключимся от сессии клиентской машины:

```
# exit
```

```
$ exit
```

Запустим сканирование клиентской машины:

```
# oscap-ssh user@192.168.10.20 22 oval eval --results oval-ssh-results.xml  
--report oval-ssh-results.html /usr/share/openscap/cpe/openscap-cpe-oval.xml
```

Откроем файл *oval-ssh-results.html*:

```
# firefox oval-ssh-results.html
```

File:///root/.oval-ssh-results.html

File:///root/.oval-ssh-results.html

OVAL Results Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	cpe:/a:open-scap:oscap	1.3.10	2024-11-12	00:00:54
#x	#v	#Error	#Unknown	#Other
0	0	0	7	53

OVAL Definition Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	vim		2012-11-22	15:00:00
#Definitions	#Tests	#Objects	#States	#Variables
60 Total	67	16	64	0

System Information

Host Name	alt-2	
Operating System	ALT Workstation	
Operating System Version	10.2	
Architecture	x86_64	
Interfaces	Interface Name	lo
	IP Address	127.0.0.1
	MAC Address	00:00:00:00:00:00
	Interface Name	enp0s3
	IP Address	10.0.2.15
	MAC Address	08:00:27:06:C7:CC
	Interface Name	enp0s8
	IP Address	192.168.10.20
	MAC Address	08:00:27:AB:FE:32
	Interface Name	lo
	IP Address	-1
	MAC Address	00:00:00:00:00:00
	Interface Name	enp0s3
	IP Address	fe00::a00:27ff:fe06:c7cc
	MAC Address	08:00:27:06:C7:CC
	Interface Name	enp0s3
	IP Address	fe80::a00:27ff:fe06:c7cc
	MAC Address	08:00:27:06:C7:CC
	Interface Name	enp0s8
	IP Address	fe80::a00:27ff:feab:fe32
	MAC Address	08:00:27:AB:FE:32

OVAL System Characteristics Generator Information

Schema Version	Product Name	Product Version	Date	Time
5.10.1	cpe:/a:open-scap:oscap		2024-11-12	00:00:54

OVAL Definition Results

ID	Result	Class	Reference ID	Title
----	--------	-------	--------------	-------

File:///root/.oval-ssh-results.html

File:///root/.oval-ssh-results.html

File:///root/.oval-ssh-results.html

File:///root/.oval-ssh-results.html

## ЗАДАНИЕ

Загрузите *Podman*-образ *BusyBox* и запустите его. С помощью *OpenSCAN* выполните сканирование на уязвимости запущенного *Podman*-образа. Сформируйте отчет в формате *html*.

## 2. AIDE

*AIDE* (Advanced Intrusion Detection Environment — усовершенствованная система обнаружения вторжений) используется для защиты от вредоносных программ, вирусов и обнаружения несанкционированных действий. Для проверки целостности файлов и обнаружения вторжений *AIDE* создает базу данных с информацией о файлах и сравнивает текущее состояние системы с этой базой. *AIDE* помогает сократить время расследования инцидентов, сосредоточившись на файлах, которые были изменены.

Установим пакет *aide*:

```
# apt-get install aide
```

Выведем версию *AIDE*:

```
# aide --version
```

или

```
# aide -v
```

```
[root@alt-1 ~]# aide --version
JAIDE 0.18.8

Compile-time options:
use pcre2: mandatory
use pthread: yes
use zlib compression: yes
use POSIX ACLs: yes
use SELinux: yes
use xattr: yes
use POSIX 1003.1e capabilities: yes
use e2fsattrs: yes
use cURL: yes
use Mhash: no
use GNU crypto library: yes
use Linux Auditing Framework: yes
use locale: no
syslog ident: aide
syslog logopt: LOG_CONS
syslog priority: LOG_NOTICE
default syslog facility: LOG_LOCAL0
```

Версия системы *AIDE* 0.18.8.

Сначала необходимо создать базу данных (снимок) всех файлов и каталогов сервера:

```
# aide --init
```

**Примечание!** Выполнение команды может занять некоторое время в зависимости от размера файловой системы и объема оперативной памяти на сервере.

**Важно!** После установки пакета необходимо исправить синтаксические ошибки в конфигурационном файле `/etc/aide.conf`:

1. Замените все табы на пробелы.
2. Начиная с 64-й строки, замените запятые на плюсы.
3. Начиная с 84 строки, уберите знаки "равно" и напротив `/dev/pts` укажите `DEVICES`.
4. Закомментируйте 87-ю строку.

Результат выполнения команды:

```
[root@alt-1 ~]# aide --init
Start timestamp: 2024-11-12 01:00:17 +0300 (AIDE 0.18.8)
AIDE successfully initialized database.
New AIDE database written to /var/lib/aide/aide.db.new.gz

Number of entries:      330253

-----
The attributes of the (uncompressed) database(s):
-----

/var/lib/aide/aide.db.new.gz
SHA256      : MPbKzmIfTm17G7JVC3ZhUojCXCKFG02n
              Ss/op0Nga50=
SHA512      : 571oSyjwAEZ/L4RoE17IEy341JveiDTE
              tJxVpmjimpdM90gPE0u5wwZcnAd6HLMdm
              BEvxYE6ZdDC3ELXdR/yUAQ==
CRC32       : OdR22Q==
GOST        : x84TTnDFLutFjq8m/ZC0H3Bw5+P3xC9o
              GTQF8HfMURo=
STRIBOG256  : Krw2rXVoIA08R3Y5xtzF2VgY3FfzOX9S
              7VLP77XDGJ0=
STRIBOG512  : f4AdH5z/d/3PWJqmyr+PsqS1MHnUXxif
              zCtfAukb8bGbVniNG5wJQq0VChRbqHyi
              l4uYb/Q4aABhkIeIQUyX1Q==

End timestamp:          (run time: 4m 1s)
```

*AIDE* не будет использовать файл *aide.db.new.gz*, пока его не переименовать *aide.db.gz*:

```
# mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

Рекомендуется периодически обновлять эту базу данных, чтобы обеспечить необходимый мониторинг изменений.

Для изменения местоположения базы необходимо изменить параметр *DBDIR* в файле */etc/aide.conf*.

Теперь *AIDE* готова к использованию новой базы данных.

Запустим первую проверку *AIDE*:

```
# aide --check > aide_check.txt
```

В результате состояние системы будет сравнено с эталонной базой данных. Все расхождения будут показаны в отчете:

```
# less report.txt
```



```

Start timestamp: (AIDE 0.18.8)
AIDE found differences between database and filesystem!!

Summary:
Total number of entries:    330253
Added entries:              330253
Removed entries:           0
Changed entries:            0

-----
Added entries:
-----

d+++++: /bin
l+++++: /bin/arp
f+++++: /bin/ash.static
l+++++: /bin/awk
f+++++: /bin/basename
l+++++: /bin/bash
f+++++: /bin/bash4
l+++++: /bin/bunzip2
l+++++: /bin/bzcat
f+++++: /bin/bzip2

```

Отчет содержит информацию о файлах и директориях, где первый символ в строке указывает на тип элемента (f — файл, d — директория). Первая секция представляет собой добавленные элементы, где тип файла сопровождается набором плюсов, указывающим на то, что все остальные атрибуты были добавлены в базу данных.

Ниже представлен список изменений, где знак равенства (=) обозначает, что файл не изменился; знак больше (>) указывает на увеличение размера файла, а знак меньше (<) — на уменьшение. Буквы в этом списке обозначают измененные атрибуты:

m и c относятся к времени изменения (mtime) и времени изменения метаданных (ctime) соответственно;

Н указывает на контрольные суммы, а b — на количество блоков.

Если атрибут был добавлен, на его месте будет стоять плюс (+), если удален — минус (-), если игнорируется — двоеточие (:), не проверен — пробел, а не изменился — точка (.).

При прокрутке вывода ниже можно получить подробный отчет по каждому файлу с указанием всех изменившихся реквизитов:

```
-----
The attributes of the (uncompressed) database(s):
-----

/var/lib/aide/aide.db.gz
SHA256      : 47DEQpj8HBSa+/TImW+5JCeuQeRkm5NM
             pJWZG3hSuFU=
SHA512      : z4PhNX7vuL3xVChQ1m2AB9Yg5AULVxXc
             g/SpIdNs6c5H0NE8XYXysP+DGNKHfuwv
             Y7kxvUdBeoG10DJ6+SfaPg==
CRC32       : AAAAAA==
GOST        : zoW5nMRnUv/+41yrmnsCeKu0wtIFXP9o
             WvSRLE1JD40=
STRIBOG256  : P10aIT6XyALMip1HTGqjKoJaNgsqkzqU
             n9klII2c4bs=
STRIBOG512  : jpRdogmqhp8EVZKFKbyuRnnphzq3B7VT
             FfVs65i+8Kc2L3FVKDVu6DzaXyqsTGrS
             ujpXbVNgcu0n5C/TBwaig==

End timestamp:  (run time: 8m 13s)
```

В целом, представленная информация позволяет как быстро просмотреть список изменений, так и провести детальное расследование.

Если оставить все как есть, файлы будут продолжать отображаться во всех последующих отчетах, поскольку их состояние отличается от указанного в эталонной базе данных. Для фиксации изменений следует обновить эталонную базу.

Для этого выполните сканирование с использованием ключа *--update*:

```
# aide --update > aide_update.txt
```

Данная команда выполняет ту же операцию, что и *--check*, однако записывает текущее состояние системы в файл базы данных *aide.db.new.gz*. После завершения процесса достаточно переименовать этот файл в *aide.db.gz*, чтобы использовать текущее состояние как новый эталон:

```
# mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

В дальнейшем рекомендуется выполнять сканирования сразу с ключом *--update*. Также настоятельно рекомендуется обновлять эталонную базу после внесения изменений в правила, особенно если были добавлены или исключены какие-либо расположения.

## ЗАДАНИЕ

Используя *AIDE* выполните проверку целостности файлов с ограничением по каталогу */etc* и с сохранением отчета в файл *check\_etc.txt*.

### 3. Maldet

*LMD (Linux Malware Detect)*, часто известный под названием *Maldet*, представляет собой сканер вредоносных программ для ОС на базе *Linux*. Распространяется под лицензией *GNU*.

*Maldet* является эффективным инструментом для обнаружения вредоносных файлов, так как включает в себя базу данных, специально разработанную для работы в среде виртуального хостинга. Его внедрение не требует значительных усилий со стороны системного администратора, что делает его удобным для использования.

Установим пакет *maldet*:

```
# apt-get install maldet
```

Выведем версию *Maldet*:

```
# maldet --version
```

или

```
# maldet -v
```

```
[root@alt-1 ~]# maldet --version
Linux Malware Detect v1.5
      (C) 2002-2015, R-fx Networks <proj@rfxn.com>
      (C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2

signature set: 2015112028602
usage maldet [-h|--help] [-a|--scan-all PATH] [-r|--scan-recent PATH DAYS]
[-f|--file-list PATH] [-i|--include-regex] [-x|--exclude-regex]
[-b|--background] [-m|--monitor] [-k|--kill-monitor] [-c|--checkout]
[-q|--quarantine] [-s|--restore] [-n|--clean] [-l|--log] [-e|--report]
[-u|--update-sigs] [-d|--update-ver]
```

Версия сканера *Maldet* 1.5.

Конфигурационный файл *Maldet* расположен по пути */etc/maldetect/conf.maldet*.

```
# cat /etc/maldetect/conf.maldet
```

Обратите внимание на параметры, касающиеся карантина, так как вы можете настроить *Maldet* на перемещение зараженных или подозрительных файлов в карантинный каталог.

Запустим проверку каталога */home/user*:

```
# maldet -a /home/user
```

```
[root@alt-1 ~]# maldet -a /home/user
Linux Malware Detect v1.5
      (C) 2002-2015, R-fx Networks <proj@rfxn.com>
      (C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2

maldet(34287): {scan} signatures loaded: 10822 (8908 MD5 / 1914 HEX / 0 USER)
maldet(34287): {scan} building file list for /home/user, this might take awhile...
maldet(34287): {scan} setting nice scheduler priorities for all operations: cpunice 19 , ionice 6
maldet(34287): {scan} file list completed in 0s, found 183 files...
maldet(34287): {scan} scan of /home/user (183 files) in progress...
maldet(34287): {scan} 183/183 files scanned: 0 hits 0 cleaned
maldet(34287): {scan} scan completed on /home/user: files 183, malware hits 0, cleaned hits 0, time 17s
maldet(34287): {scan} scan report saved, to view run: maldet --report 241112-0214.34287
```

Когда сканирование будет завершено, сканер сообщит какой *SCANID* был присвоен отчёту (в примере 241112-0214.34287).

Откроем отчет сканирования каталога */home/user*:

**# maldet -e 241112-0214.34287**

```
GNU nano 7.2 /var/lib/maldetect/sess/session.241112-0214.34287
HOST: alt-1
SCAN ID: 241112-0214.34287
STARTED:
COMPLETED:
ELAPSED: 17s [find: 0s]

PATH: /home/user
TOTAL FILES: 183
TOTAL HITS: 0
TOTAL CLEANED: 0

=====
Linux Malware Detect v1.5 < proj@rfxn.com >

Описание
Справка

^G Справка  ^O Записать  ^W Поиск     ^K Вырезать  ^T Выполнить ^C Позиция
^X Выход     ^R ЧитФайл   ^_ Замена    ^U Вставить  ^J Выровнять ^_/ К строке
```

Для работы мониторинга используется библиотека *inotify-tools*.

Установим пакет *inotify-tools*:

**# apt-get install inotify-tools**

Мониторинг можно настроить на отдельного пользователя, каталог или даже отдельный файл. Он будет отслеживать изменение или появление новых файлов по указанному пути и запускать их проверку.

Запустим мониторинг каталога */home/user*:

**# maldet -m /home/user**

```
[root@alt-1 ~]# maldet -m /home/user/
Linux Malware Detect v1.5
(C) 2002-2015, R-fx Networks <proj@rfxn.com>
(C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2

maldet(37842): {mon} added /home/user to inotify monitoring array
maldet(37842): {mon} added /dev/shm to inotify monitoring array
maldet(37842): {mon} added /var/tmp to inotify monitoring array
maldet(37842): {mon} added /tmp to inotify monitoring array
maldet(37842): {mon} starting inotify process on 4 paths, this might take awhile...
maldet(37842): {mon} inotify startup successful (pid: 38409)
maldet(37842): {mon} inotify monitoring log: /var/lib/maldetect/logs/inotify_log
```

Статистику мониторинга можно отслеживать в журнале `/var/lib/maldetect/logs/inotify_log`:

```
# tail -f /var/lib/maldetect/logs/inotify_log
```

Откроем еще одну вкладку терминала MATE.

Создадим файл `test` в домашнем каталоге пользователя `user`:

```
$ touch test
```

```
[user@alt-1 ~]$ touch test
[user@alt-1 ~]$
```

В результате в журнале появится следующая запись:

```
[root@alt-1 ~]# tail -f /var/lib/maldetect/logs/inotify_log
/home/user/test CREATE
```

Полностью остановить сервис мониторинга при необходимости можно с помощью следующей команды:

```
# maldet -k
```

Обновим базы вирусных сигнатур:

```
# maldet -u
```

```
[root@alt-1 ~]# maldet -u
Linux Malware Detect v1.5
  (C) 2002-2015, R-fx Networks <proj@rfxn.com>
  (C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2

maldet(39617): {sigup} performing signature update check...
maldet(39617): {sigup} local signature set is version 2015112028602
maldet(39617): {sigup} new signature set (202411101345636) available
maldet(39617): {sigup} downloading http://cdn.rfxn.com/downloads/maldet-sigpack.tgz
maldet(39617): {sigup} downloading http://cdn.rfxn.com/downloads/maldet-cleanv2.tgz
maldet(39617): {sigup} verified md5sum of maldet-sigpack.tgz
maldet(39617): {sigup} unpacked and installed maldet-sigpack.tgz
maldet(39617): {sigup} verified md5sum of maldet-clean.tgz
maldet(39617): {sigup} unpacked and installed maldet-clean.tgz
maldet(39617): {sigup} signature set update completed
maldet(39617): {sigup} 16855 signatures (14801 MD5 / 2054 HEX / 0 USER)
```

Если вы не настроили отправку инфицированных файлов в карантин через конфигурационный файл, эту операцию можно выполнить вручную.

Переместим все инфицированные файлы, обнаруженные в ходе проверки с идентификатором `241112-0214.34287` в каталог `/var/lib/maldetect/quarantine`:

```
# maldet -q 241112-0214.34287
```

```
[root@alt-1 ~]# maldet -q 241112-0214.34287
Linux Malware Detect v1.5
  (C) 2002-2015, R-fx Networks <proj@rfxn.com>
  (C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2
```

Также можно попытаться автоматически вылечить все файлы, помещенные в карантин:

```
# maldet -n 241112-0214.34287
```

```
[root@alt-1 ~]# maldet -n 241112-0214.34287
Linux Malware Detect v1.5
  (C) 2002-2015, R-fx Networks <proj@rfxn.com>
  (C) 2015, Ryan MacDonald <ryan@rfxn.com>
This program may be freely redistributed under the terms of the GNU GPL v2
```

Если после проверки вы заключили, что лечение прошло успешно, восстановить файлы из карантина:

```
# maldet -s 241112-0214.34287
```

### **ЗАДАНИЕ**

1. Настройте сканер *Maldet* так, чтобы он игнорировал каталог */etc/openssh*. Убедитесь, что файлы из этой директории не проверяются при сканировании.

2. Настройте сканер *Maldet* так, чтобы он игнорировал файлы с расширениями *.bak* и *.log*. Убедитесь, что такие файлы не проверяются при сканировании.