Dr.WEB

vxCube

Руководство администратора

Dr.Web vxCube
Версия 1.6.0
Руководство администратора
07.03.2025

ООО «Доктор Веб», Центральный офис в России

Адрес: 125124, Москва, ул. 3-я Ямского Поля, д. 2, корп. 12А

Сайт: https://www.drweb.com/

Телефон: +7 (495) 789-45-87

Информацию о региональных представительствах и офисах вы можете найти на официальном сайте компании.

ООО «Доктор Веб»

Компания «Доктор Веб» — российский разработчик средств информационной безопасности.

Компания «Доктор Веб» предлагает эффективные антивирусные и антиспам-решения как для государственных организаций и крупных компаний, так и для частных пользователей.

Антивирусные решения семейства Dr.Web разрабатываются с 1992 года и неизменно демонстрируют превосходные результаты детектирования вредоносных программ, соответствуют мировым стандартам безопасности.

Сертификаты и награды, а также обширная география пользователей свидетельствуют об исключительном доверии к продуктам компании.

Мы благодарны пользователям за поддержку решений семейства Dr.Web!

Содержание

1. Условные обозначения

В данном руководстве используются следующие условные обозначения:

Обозначение	Комментарий
	Предупреждение о возможных ошибочных ситуациях, а также важных моментах, на которые следует обратить особое внимание.
Антивирусная сеть	Новый термин или акцент на термине в описаниях.
<IP-address>	Поля для замены функциональных названий фактическими значениями.
Сохранить	Названия экранных кнопок, окон, пунктов меню и других элементов программного интерфейса.
CTRL	Обозначения клавиш клавиатуры.
`C:\Windows\`	Наименования файлов и каталогов, фрагменты программного кода.
Приложение А	Перекрестные ссылки на главы документа или гиперссылки на внешние ресурсы.

2. О продукте

Dr.Web vxCube — это сервис, который анализирует потенциально вредоносные файлы и формирует подробный отчет об их поведении в заданных условиях.

Для анализа Dr.Web vxCube использует техники *аппаратной виртуализации*. Это позволяет Dr.Web vxCube работать быстро и оставаться невидимым для анализируемого файла.

Вы можете загрузить любой из поддерживаемых типов файлов в анализатор, выбрать условия, в которых он будет исполняться на виртуальной машине, и влиять на ход выполнения анализа. После проверки вы получите полный технический отчет, а также видеоотчет о поведении файла в заданных условиях.

2.1. Особенности Dr.Web vxCube

Ниже приведены основные особенности сервиса Dr.Web vxCube:

- Виртуальные машины подключаются к интернету через выделенный прокси-сервер. Это позволяет анализировать поведение файла в полном объеме, особенно если его работа напрямую зависит от загрузки данных из сети.
- Новый механизм анализатора работает на уровне *гипервизора* и не использует дополнительное программное обеспечение (например, специальные драйверы для перехвата функций) в гостевой операционной системе. Это не позволяет исследуемому образцу обнаруживать или снимать перехваты.
- Журнал событий ведется на уровне гипервизора, поэтому обнаружить анализатор невозможно.
- К анализируемой среде можно подключиться с помощью VNC-клиента (Virtual Network Computing) и влиять на процесс анализа.

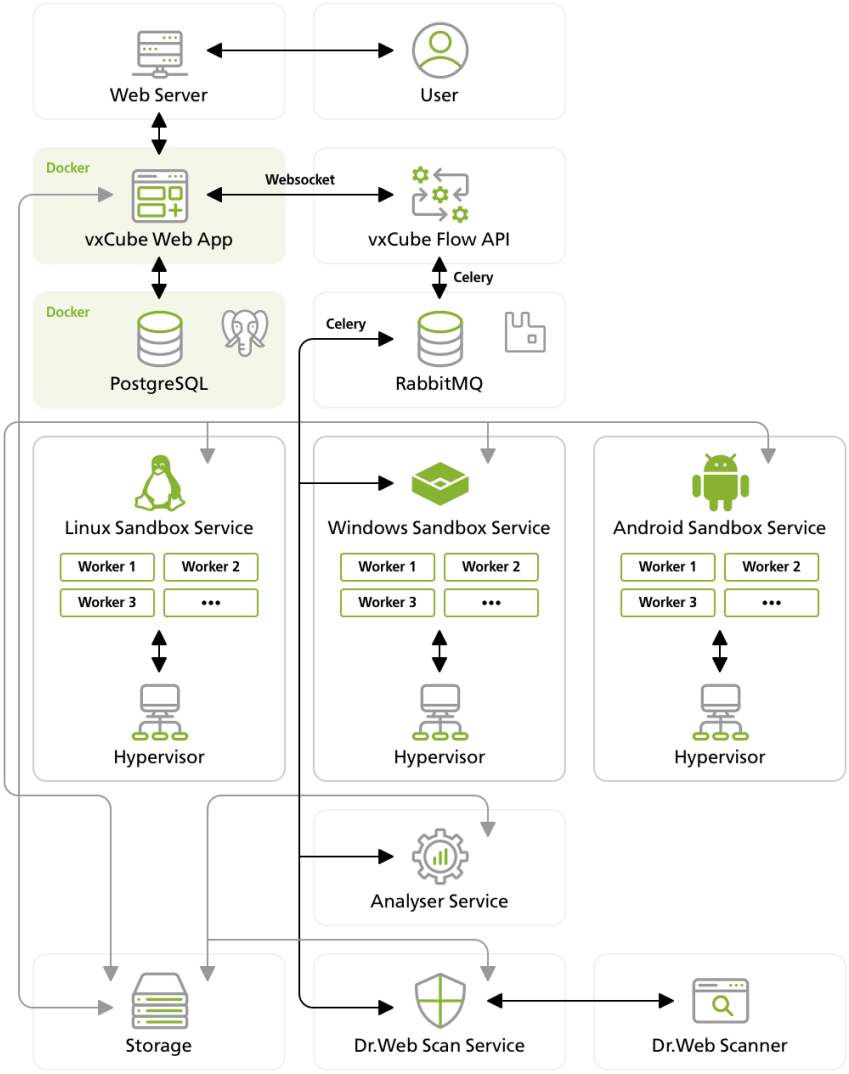
2.2. Как работать с Dr.Web vxCube

Чтобы проверить подозрительный файл на угрозы с помощью Dr.Web vxCube, выполните следующие действия:

1. Загрузите файл, который нужно проверить, в Dr.Web vxCube.
2. (Необязательно) Укажите дополнительные настройки анализа и запустите анализ.
3. Изучите отчет, сформированный сервисом Dr.Web vxCube по результатам проверки.

2.3. Как устроен Dr.Web vxCube

Продукт состоит из нескольких компонентов и сервисов, взаимодействующих между собой. Схема устройства представлена ниже.



vxCube Web App

Основное приложение, предоставляющее удобный интерфейс для взаимодействия с системой анализа файлов. Также предоставляет API для автоматизации задач анализа файлов. Для удобства использования API реализована библиотека на Python.

vxCube Flow API

Компонент, отвечающий за логику распределения заданий по анализу файлов между различными сервисами. Позволяет удобно добавлять новые сервисы в систему анализа.

Windows Sandbox Service
Сервис запуска файлов в специальной виртуальной среде ОС Windows. Виртуальная машина реализована в виде модифицированного гипервизора с установкой перехватов и применением техник аппаратной виртуализации.

Linux Sandbox Service
Сервис динамического анализа ELF-файлов. Файлы запускаются на виртуальной машине с соответствующей архитектурой и разрядностью, и действия файла логирует специальный драйвер, установленный на виртуальной машине.

Если все компоненты, входящие в состав vxCube, будут установлены на одном сервере, может наблюдаться замедление работы сервиса Linux Sandbox Service. Это связано с тем, что виртуализация на нем реализуется программно. Поэтому рекомендуем устанавливать каждый компонент на отдельный сервер.

Android Sandbox Service (опционально)
Сервис запуска файлов в специализированной виртуальной среде Android, представленный в виде уникальной реализации образа ОС Android.

Analyser Service
Сервис анализа поведения, записанного в виртуальной машине. В нем оценивается вредоносность поведения файла, а также формируются описания (текстовое, MAEC, STIX).

Dr.Web Scan Service
Сервис сканирования файлов и дампов памяти, полученных в процессе запуска образца.

Передача файлов между компонентами осуществляется через общее хранилище (на схеме **Storage**), реализованное в виде FTP-сервера. Для хранения данных о пользователях и результатах анализа используется база данных *PostgreSQL*. Для передачи заданий между сервисами используется брокер сообщений *RabbitMQ*.

Чтобы обеспечить максимальную безопасность, каждая виртуальная машина имеет свое собственное изолированное сетевое пространство. Для доступа в интернет используется VPN-сервер. Для корректной работы VPN-сервер должен быть сконфигурирован самостоятельно. Более подробная информация приведена в Приложении В. Настройка отдельного VPN-сервера.

Рекомендуется использовать выделенный VPN-сервер, поскольку публичные сервисы могут быть сконфигурированы некорректно.

2.4. Системные требования

Для установки Dr.Web vxCube требуется компьютер, удовлетворяющий следующим требованиям:

Параметр	Требования
Процессор	Intel с поддержкой технологий Intel VT-x, EPT и Preempt Timer Минимально: 16 ядер Желательно: 48 ядер Оптимально: 56 ядер
Оперативная память	Минимально: 64 ГБ Желательно: 128 ГБ Оптимально: 256 ГБ
Накопитель	Минимально: 1 SSD-диск объемом 256 ГБ Желательно: 2 SSD-диска по 256 ГБ каждый Оптимально: 4 SSD-диска по 256 ГБ каждый Чтобы избежать проблем в работе SSD-дисков, убедитесь, что они подключены через интерфейс SATA 3 и не используются в RAID-массивах.
Операционная система	• Astra Linux 1.7.6 с ядром Linux 5.4 (должна быть установлена на отдельном HDD- или SSD-диске). • Ubuntu 22.04 с ядром Linux 5.15. SWAP-раздел должен составлять не менее 10–50 % от объема оперативной памяти.
Интеграция с MailD	Требуется наличие SSL-сертификатов для работы по протоколу HTTPS. Без этого vxCube не сможет проверять письма в формате EML.

Для комфортной работы с веб-интерфейсом Dr.Web vxCube требуются:

Параметр	Требования
Браузер	• Google Chrome версии 60.0 и более поздних. • Mozilla Firefox версии 55.0 и более поздних. • Safari версии 11.0 и более поздних. • Opera версии 47.0 и более поздних. В Windows XP рекомендуется использовать браузер Google Chrome. Кроме того, в Windows XP не гарантируется воспроизведение видео в браузере Mozilla Firefox.
Разрешение экрана	Не менее 1024x768 пикселей.
Дополнительно	Если вы хотите управлять процессом анализа в интерактивном режиме,

Параметр	Требования
	убедитесь, что в вашем браузере разрешено открытие всплывающих окон.

3. Как установить Dr.Web vxCube

Перед началом установки убедитесь, что компьютер, с которого вы будете устанавливать Dr.Web vxCube, соответствует системным требованиям.

Для установки используется технология Ansible Playbook, которая позволяет устанавливать программное обеспечение и настраивать окружение для нескольких устройств одновременно.

3.1. Подготовка к установке

После того как вы приобретете лицензию на Dr.Web vxCube, на адрес вашей электронной почты придут следующие письма:

- со ссылками на дистрибутив продукта и образы виртуальных машин, на которых будет производиться анализ;
- с лицензионным ключевым файлом.

Кроме того, отдельной посылкой вы получите аппаратный ключ защиты Guardant.

Прежде чем приступить к установке, выполните указанные ниже действия:

1. Очистите SSD-диск, на который вы устанавливаете компоненты Dr.Web vxCube. Если SSD-диск разбит на разделы, удалите их.
2. Подключите электронный ключ защиты Guardant к устройству, на котором будет развернут сервис Dr.Web vxCube. Это обязательно для запуска анализа. Если вы будете разворачивать Dr.Web vxCube на нескольких устройствах, электронный ключ защиты Guardant потребуется для каждого из них.
3. Скачайте дистрибутив, образы и ключевой файл из полученных писем. Распакуйте скачанные архивы.
4. Поместите лицензионный ключ Dr.Web vxCube в `confs/vxcube.key`.
5. Поместите файлы дистрибутива и образов в следующие каталоги:
   - SSL-сертификаты для подключения к VPN-серверам, указанным в файле `vars-user.yml` (переменная `openvpn_client_servers`), в каталоги `confs/openvpn.crt`, `confs/openvpn.key` и `confs/openvpn_ca.crt`;

     SSL-сертификаты не являются частью продукта Dr.Web vxCube; их необходимо приобрести отдельно.

   - полученные образы виртуальных машин на базе Windows с расширением `.tar.gz` и все служебные файлы `.tar.gz.ver` и `.tar.gz.hash` в каталог `vm-images-win`;
   - полученные образы виртуальных машин на базе Android с расширением `.vdi` и все служебные файлы `.vdi.hash` и `.vdi.ver` в каталог `vm-images-andr`;

- полученные образы виртуальных машин на базе Linux с расширением `.tar.gz` и все служебные файлы `.tar.gz.hash` и `.tar.gz.ver` в каталог `vm-images-linux`.

Вы также можете поместить образы виртуальных машин в произвольные каталоги. В этом случае вам нужно указать эти каталоги в файле `vars-default.yml` (в переменных `hyperbox_images_repo`, `dimas_images_repo` и `linuxbox_images_repo`).

6. Если вы хотите интегрировать Dr.Web vxCube с MailD, вам потребуется SSL-сертификат. Если у вас нет этого сертификата, его можно сгенерировать, например с помощью следующей команды:

```
openssl dhparam -out web_dhparam.pem 2048
```

7. Если вы хотите, чтобы веб-интерфейс Dr.Web vxCube поддерживал работу по протоколу HTTPS, выполните следующие действия:
   - создайте файлы `confs/web_ssl.crt`, `confs/web_ssl.key`;
   - создайте файл `confs/web_dhparam.pem` (если пропустили шаг 5);
   - в файле `vars-default.yml` раскомментируйте переменные `vxcube_web_ssl_cert`, `vxcube_web_ssl_privkey` и `vxcube_web_dhparam`.
8. Убедитесь, что на всех серверах, на которых вы собираетесь развертывать компоненты Dr.Web vxCube, установлены пакеты `make`, `python3-venv` и `sshpass`. Если пакетов нет, установите их, выполнив следующие команды:

```
sudo apt-get install make
sudo apt-get install python3-venv
sudo apt-get install sshpass
```

При запуске команды `make` создается виртуальное окружение Python и устанавливаются пакеты, необходимые для запуска Ansible нужной версии. Если вы используете любой другой репозиторий Python, помимо `pipy.org`, укажите его через переменные окружения, зарегистрировав их, например:

```
export PIP_INDEX_URL=https://devpi.local
export PIP_TRUSTED_HOST=devpi.local
```

Или укажите в команде запуска:

```
PIP_INDEX_URL=https://devpi.local
PIP_TRUSTED_HOST=devpi.local make deploy
```

9. Включите поддержку `systemd-networkd.socket` с помощью следующих команд:

```
sudo systemctl stop systemd-networkd.service
sudo systemctl enable systemd-networkd.socket
sudo systemctl start systemd-networkd.socket
sudo systemctl start systemd-networkd.service
```

10. (Только для установки на компьютер с ОС Astra Linux) Обновите ОС Astra Linux до версии 1.7.6. В `/etc/apt/sources.list` должны быть указаны следующие репозитории:

```
# Расширенный репозиторий

deb https://dl.astralinux.ru/astra/stable/1.7_x86-64/repository-extended/ 1.7_x86-64 main contrib non-free

# Репозиторий Astra 1.7.6

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-main/ 1.7_x86-64 main contrib non-free

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-update/ 1.7_x86-64 main contrib non-free

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-base/ 1.7_x86-64 main contrib non-free
```

11. После обновления репозиториев выполните команды:

```
sudo apt update
sudo apt upgrade
```

12. Отредактируйте настройки установки. Для этого в файле `vars-user.yml` задайте значения всех переменных. Их подробное описание вы найдете в самом файле, а также в следующей таблице:

Описание переменных в файле vars-user.yml

Переменная	Описание
`vxcube_web_superuser_email`	E-mail администратора Dr.Web vxCube. Пользователь с указанным вами адресом будет создан после установки веб-интерфейса.

`vxcube_web_superuser_pass`	Пароль администратора.
	Вы можете ввести нужный вам пароль:
	`vxcube_web_superuser_pass: "example_password"`
	Или указать его генерацию автоматически:
	`vxcube_web_superuser_pass: "{{ lookup('password', 'credentials/vxcube_web_superuser_pass length=10 chars=ascii_letters,digits') }}"`
	Сгенерированный пароль будет сохранен в файле `credentials/vxcube_web_superuser_pass`.

`hyperbox_ssds`	Список доступных на сервере SSD-дисков. Пример значения переменной для двух дисков: `hyperbox_ssds:`

Переменная	Описание
	- sdb - sdc
vxcube_os_count	Максимальное число работающих одновременно виртуальных машин с ОС Windows. При выборе значения для этой переменной необходимо учитывать технические характеристики устройства. Оптимальное значение можно рассчитать по формуле: *<vxcube_os_count> = <количество ядер * 1.5> / <количество разных типов ОС>*. К примеру, на 48-ядерном сервере при использовании четырех различных типов ОС можно установить значение vxcube_os_count, равное 18.
vxcube_os_clone_threads	Число потоков, используемых для клонирования виртуальных машин. При выборе значения для этой переменной учитывайте технические характеристики устройства. К примеру, при наличии двух SSD-дисков и 48-ядерного процессора количество потоков не должно превышать 15.
openvpn_client_servers	Список VPN-серверов, через которые будет маршрутизироваться трафик с виртуальных машин. Если в переменной указано несколько хостов, при выходе из строя первого сервера будет использоваться следующий по списку. Обратите внимание, что vxcube-installer не развертывает VPN-сервер. Для этого вам необходимо настроить VPN-сервер самостоятельно. Подробнее см. Приложение В. Настройка отдельного VPN-сервера. Формат значения переменной: openvpn_client_servers: - host: xx.xx.xx.xx port: 1194
vboxnet_vpn_gateway	IP-адрес шлюза внутри VPN-сети, на который будет перенаправляться трафик. Как правило, это будет первый хост в подсети. К примеру, если внутренняя VPN-сеть имеет подсеть 10.0.42.0/24, то шлюзом будет являться 10.0.42.1.
evparser_max_workers_count	Максимальное количество рабочих потоков сервиса анализа поведения. При выборе значения для этой переменной учитывайте технические характеристики устройства. Рекомендуется установить значение, равное переменной vxcube_os_count.
evparser_min_workers_count	Минимальное количество рабочих потоков сервиса анализа поведения. Они будут запущены всегда, даже при отсутствии заданий. Рекомендуется установить значение, равное 20 % от evparser_max_workers_count.

Переменная	Описание
`evparser_srv_autoscale`	Максимальное и минимальное количество рабочих потоков сервиса анализа поведения. Формат значения переменной: `"<max_worker_num>,<min_worker_num>"` Например: `evparser_srv_autoscale: "10,1"`
`drweb_srv_autoscale`	Максимальное и минимальное количество рабочих потоков, используемых антивирусом для проверки файлов. Формат значения переменной: "*<max_worker_num>*,*<min_worker_num>*". Например: `drweb_srv_autoscale: "10,1"` Рекомендуется установить значение `max_worker_num`, равное значению `vxcube_os_count`, а `min_worker_num`, равное 20 % от значения `max_worker_num`.
`yara_srv_autoscale`	Максимальное и минимальное количество рабочих потоков, используемых сервисом YARA. Формат значения переменной: *<max_worker_num>*,*<min_worker_num>*. Например: `yara_srv_autoscale: "10,1"` Рекомендуется установить значение `max_worker_num`, равное значению `vxcube_os_count`, а `min_worker_num`, равное 20 % от значения `max_worker_num`.

Дополнительно вы также можете указать или изменить значения переменных в файле `vars-default.yml`. Их подробное описание вы найдете в самом файле, а также в следующей таблице:

Описание переменных в файле vars-default.yml

Переменная	Описание
`vxcube_local_hostname`	Имя хоста при загрузке на локальный сервер.
`vxcube_ftp_user`	Логин FTP-пользователя, используется для загрузки образцов на FTP-сервер. Например: `vxcube_ftp_user: "vxcube_ftp"`
`vxcube_ftp_pass`	Пароль FTP-пользователя, указанного в переменной `vxcube_ftp_user`. По умолчанию генерируется случайный пароль, который записывается в файл `credentials/vxcube_ftp_pass`. Если пароль уже был сгенерирован ранее (то есть файл `credentials/vxcube_ftp_pass` уже создан и пароль туда записан), будет использоваться пароль из файла. Например:

Переменная	Описание
	`vxcube_ftp_pass: "{{ lookup('password', 'credentials/vxcube_ftp_pass length=10 chars=ascii_letters,digits') }}"`
`vxcube_web_db_pass`	Пароль для базы данных. По умолчанию генерируется пароль, который записывается в файл `credentials/vxcube_web_db_pass`. Например: `vxcube_web_db_pass: "{{ lookup('password', 'credentials/vxcube_web_db_pass length=10 chars=ascii_letters,digits') }}"`
`vxcube_web_ssl_cert`	Путь до сертификата `.crt`. По умолчанию сертификат находится в `confs/web_ssl.crt`. Например: `vxcube_web_ssl_cert: "{{ lookup('file', 'confs/web_ssl.crt') }}"`
`vxcube_web_ssl_privkey`	Путь до приватного ключа `.key`. По умолчанию ключ находится в `confs/web_ssl.key`. Например: `vxcube_web_ssl_privkey: "{{ lookup('file', 'confs/web_ssl.key') }}"`
`vxcube_web_dhparam`	Путь до ключа Диффи-Хеллмана `.pem`. По умолчанию ключ находится в `confs/web_dhparam.pem`. Например: `vxcube_web_dhparam: "{{ lookup('file', 'confs/web_dhparam.pem') }}"`
`vxcube_web_mail_server`	IP-адрес или доменное имя SMTP-сервера. Например: `vxcube_web_mail_server: "localhost"` Значение по умолчанию: `localhost`.
`vxcube_web_recaptcha_site_key`	Ключ сайта, используется для включения reCAPTCHA. Например: `vxcube_web_recaptcha_site_key: SITE_KEY //<ключ сайта>` Значения берутся при регистрации домена на https://www.google.com/recaptcha/admin.
`vxcube_web_recaptcha_secret`	Секретный ключ, используется для включения reCAPTCHA. Например:

Переменная	Описание
	`vxcube_web_recaptcha_secret: SECRET_KEY //<секретный ключ>` Значения берутся при регистрации домена на https://www.google.com/recaptcha/admin.
`vxcube_web_check_exchange_min`	Период проверки директории FTP-сервера в минутах. Принимает диапазон значений от 1 до 60. Например: `vxcube_web_check_exchange_min: "5"`
`vxcube_web_keep_free_space_percent`	Если значение этой переменной меньше, чем объем свободного места на диске, то старые отчеты удаляться не будут. Значение в процентах. Например: `vxcube_web_keep_free_space_percent: "30"`
`vxcube_web_keep_exchange_hours`	Минимальное время хранения файлов при нехватке места на диске. Значение в часах. Если переменная имеет значение 0,5, то созданные за последние 0,5 часа файлы не будут удаляться. Например: `vxcube_web_keep_exchange_hours: "0,5"`
`vxcube_web_keep_reports`	Минимальное время после создания отчета, после которого отчет считается старым и удаляется. Значение в минутах. Например: `vxcube_web_keep_reports: "20m"`
`vxcube_web_reports_clean_period_min`	Периодичность запуска задачи на удаление старых отчетов. Значение в минутах. Например: `vxcube_web_reports_clean_period_min: "5"`
`vxcube_web_fail_free_space_percent`	Свободное место на диске, при котором запущенные задачи будут завершаться с ошибкой. Значение в процентах. Например: `vxcube_web_fail_free_space_percent: "5"`
`vxcube_web_local_max_body_size`	Внутренняя переменная. Укажите для нее значение, равное значению переменной `vxcube_web_max_body_size` ниже.
`vxcube_web_local_max_exec_time`	Максимально допустимое значение параметра **Время выполнения файла по умолчанию**, указанное в секундах. Например: `vxcube_web_local_max_exec_time: "3600"`

Переменная	Описание
vxcube_web_max_body_size	Максимальный размер файла, который можно отправить на анализ, в мегабайтах. Например: `vxcube_web_max_body_size: "2001"`
hyperbox_api_rq_username	Учетная запись для работы с RabbitMQ. Например: `hyperbox_api_rq_username: "celery"`
hyperbox_api_rq_password	Пароль для пользователя, указанного в `hyperbox_api_rq_username`. По умолчанию генерируется случайный пароль в `credentials/hyperbox_api_rq_password`. Например: `hyperbox_api_rq_password: "{{ lookup('password', 'credentials/hyperbox_api_rq_password length=10 chars=ascii_letters,digits') }}"`
hyperbox_api_rq_vhost	Имя базы данных в RabbitMQ. Например: `hyperbox_api_rq_vhost: "tasks"`
hyperbox_api_rq_admin_pass	Пароль администратора для работы с пользователями и базой данных в RabbitMQ. Например: `hyperbox_api_rq_admin_pass: "{{ lookup('password', 'credentials/hyperbox_api_rq_admin_pass length=10 chars=ascii_letters,digits') }}"`
hyperbox_api_rq_plugins	Список плагинов RabbitMQ, которые будут установлены. Например: `hyperbox_api_rq_plugins: ["rabbitmq_management"]` Переменная отключена по умолчанию.
hyperbox_key_path	Путь к лицензионному ключу. Например: `hyperbox_key_path: "confs/vxcube.key"` Путь по умолчанию: `confs/vxcube.key`.
hyperbox_external_addr	Адрес для подключения к серверу с виртуальными машинами с помощью VNC. Например:

Переменная	Описание
	`hyperbox_external_addr: "{{ hostvars[inventory_hostname]['ansible_default_ipv4']['address'] }}"` Переменная необходима для корректной работы VNC.
`hyperbox_hbsetup`	Используется для переклонирования виртуальных машин из OVA-образов. Например: `hyperbox_hbsetup: "false"` Значение `True` используется при первоначальной установке и в случае внесения изменений в OVA-образы.

`hyperbox_images_repo`	Путь к OVA-образам виртуальных машин.
	Например:
	`hyperbox_images_repo: "vm-images-win"`
	Значением переменной также может являться FTP-адрес в формате URI.
	Например:
	`hyperbox_images_repo: "ftp://user:pass@host:port/path"`

`hyperbox_images`	Список виртуальных машин и их характеристики. Все машины из списка должны быть доступны в репозитории образов, указанном в `hyperbox_images_repo`.
	Например:
	`hyperbox_images:`
	`- vm_type: 6.1.7601.17514_x86`
	`code: Win7x86`
	`count: "{{ vxcube_os_count }}"`
	`memory: 1536`
	`clone_threads: "{{ vxcube_os_clone_threads }}"`
	`cores: 2`
	При этом `cores` — количество ядер, выделенных виртуальной машине.

`hyperbox_force_clean_vms`	Позволяет принудительно удалять все использованные виртуальные машины перед клонированием новых. Значение по умолчанию: `False`.
`vm_type`	Тип виртуальной машины. Например: `vm_type: 6.1.7601.17514_x86`
`code`	Имя виртуальной машины. Например: `code: Win7x86`

Переменная	Описание
`count`	Количество клонов ОС для каждой из виртуальных машин. Например: `count: "{{ vxcube_os_count }}"` По умолчанию значение берется из переменной `vxcube_os_count`.
`memory`	Объем оперативной памяти, выделяемой для виртуальной машины при запуске. Например: `memory: "1536"`
`clone_threads`	Используемое число потоков при клонировании виртуальных машин во время установки. Например: `clone_threads: "{{ vxcube_os_clone_threads }}"` По умолчанию значение берется из переменной `vxcube_os_clone_threads`.
`openvpn_client_crt`	Путь до сертификата сервера OpenVPN. Например: `openvpn_client_crt: "{{ lookup('file', 'confs/openvpn.crt') }}"` Путь по умолчанию: `confs/openvpn.crt`
`openvpn_client_key`	Путь до закрытого ключа клиента сервера OpenVPN. Например: `openvpn_client_key: "{{ lookup('file', 'confs/openvpn.key') }}"` Путь по умолчанию: `confs/openvpn.key`
`openvpn_client_ca_crt`	Путь до сертификата ключевого центра сертификации сервера OpenVPN. Например: `openvpn_client_ca_crt: "{{ lookup('file', 'confs/openvpn_ca.crt') }}"` Путь по умолчанию: `confs/openvpn_ca.crt`
`openvpn_client_tls_auth`	Параметр `tls-auth` добавляет использование еще одной подписи HMAC к handshake-пакетам SSL/TLS, инициируя дополнительную проверку целостности. Например:

Переменная	Описание
	`openvpn_client_tls_auth: "{{ lookup('file', 'confs/openvpn_ta.crt', errors='ignore')| default(omit) }}"`
`openvpn_client_chiper`	Используемый метод шифрования. Например: `openvpn_client_chiper: "AES-128-CBC"`
`vxcube_optional_types_codes`	Опциональные типы файлов, доступные для анализа. Например: `vxcube_optional_types_codes: "moc"` Чтобы использовать переменную, необходимо установить ПО Microsoft Office в образах. По умолчанию отключено.
`drweb_srv_se_licence_key`	Путь к лицензионному ключу. Например: `drweb_srv_se_licence_key: "{{ lookup('file', 'confs/vxcube.key') }}"` Путь по умолчанию: `confs/vxcube.key`
`vxcube_web_server_name`	Доменное имя для доступа к веб-интерфейсу Dr.Web vxCube.
`vxcube_storage`	Путь к каталогу для виртуальных машин и временных файлов, созданных в процессе анализа. Например: `vxcube_storage: "/var/lib/storage"`
`vxcube_configure_firewall`	Индикатор необходимости выполнять настройку брандмауэра после завершения установки (true/false).
`evparser_dwschecker_url`	URL для доступа к сервису проверки ссылок. Доступно, если сервис подключен при покупке лицензии. Например: `evparser_dwschecker_url: "http://hostname/?key=mykey&url={0}&info=1"`
`dimas_external_ip`	Адрес для доступа к серверу с виртуальными машинами. Необходим для корректной работы VNC. Например: `dimas_external_ip: "{{ hostvars[inventory_hostname]['ansible_default_ipv4']['address'] }}"`
`dimas_vms_setup`	Индикатор необходимости пересоздания виртуальных машин (true/false).

Переменная	Описание
`vxcube_force_clean_vms`	Удаление файлов виртуальных машин.
`dimas_images_repo`	Путь к репозиторию с VDI-образами дисков виртуальных машин. Также можно указать FTP-адрес в формате URI. Например: `dimas_images_repo: "ftp://user:pass@host:port/path"`
`dimas_tar_repo`	Путь к репозиторию с vboxsdk. Также можно указать FTP-адрес в формате URI. Например: `dimas_tar_repo: "ftp://user:pass@host:port/path"`

`dimas_images`	Список виртуальных машин на базе Android и их характеристики.
	Все виртуальные машины из списка должны находиться в репозитории образов, указанном в `dimas_vdi_repo`.
	Например:
	`dimas_images:`
	`  - vm_type: Android7.1`
	`  code: Android7.1`
	`  count: 3`
	`  memory: 4072`
	`  cores: 2`
	`  clone_threads: 3`

`_vxcube_use_windows_workers: "{{ lookup('ini', 'Windows section=Settings file={{ hyperbox_key_path }}') | bool }}"`
 	
`_vxcube_use_android_workers: "{{ lookup('ini', 'Android section=Settings file={{ hyperbox_key_path }}') | bool }}"`
 	
`_vxcube_use_linux_workers: "{{ lookup('ini', 'Linux section=Settings file={{ hyperbox_key_path }}') | bool }}"`
 	
`vxcube_web_expire_date: "{{ lookup('ini', 'Expires section=Key file={{ hyperbox_key_path }}') }}"`
	Информация о лицензии.

Переменная	Описание
`vxcube_web_activation_date: "{{ lookup('ini', 'Created section=Key file={{ hyperbox_key_path }}') }}"`	

`zabbix_agent_required`	Переменная `zabbix_agent_required` для опциональной установки Zabbix-агента, который позволяет вести активный мониторинг состояния компонентов vxCube.
	
	Например:
	
	`zabbix_agent_required: "true"`
	`zabbix_agent_server: 192.168.33.30`
	`zabbix_agent_serveractive: 192.168.33.30`
	`zabbix_version: 3.0`
	
	Переменные `zabbix_agent_server` и `zabbix_agent_serveractive` должны содержать IP-адрес или доменное имя сервера Zabbix Server.
	
	Сам сервер Zabbix Server необходимо установить самостоятельно.
	
	Параметр `zabbix_version` опционален. Если его не указывать, на хост будет установлена последняя доступная версия Zabbix. Если необходимо использовать более старую версию, укажите ее номер.
	
	Например:
	
	`zabbix_version: 4.0`, `zabbix_version: 3.4` или `zabbix_version: 2.2`.

`zabbix_agent_serveractive`	Адреса серверов Zabbix для активных проверок.
`vxcube_web_flask_workers`	Число рабочих потоков веб-приложения. Например: `vxcube_web_flask_workers: 5`
`vxcube_web_flask_timeout`	Тайм-аут ответа от веб-приложения. Например: `vxcube_web_flask_timeout: 300`
`linuxbox_vms_setup`	Индикатор необходимости пересоздания виртуальных машин (true/false).

Переменная	Описание
	Например: `linuxbox_vms_setup: true`

`linuxbox_images_repo`	Путь к репозиторию, где лежат VDI-образы виртуальных машин. В качестве значения переменной также может быть указан FTP-адрес в формате URI.
	
	Например:
	
	`linuxbox_images_repo: "ftp://user:pass@host:port/path"`
	
	или
	
	`linuxbox_images_repo: "vm-images-linux"`

`linuxbox_images`	Список виртуальных машин на базе Linux и их характеристики. Все эти виртуальные машины должны быть доступны в репозитории образов, указанном в `linuxbox_vdi_repo`.
	
	Например:
	
	`linuxbox_images:`
	`  - vm_type: intel64_astra_ce_2.12`
	`  code: intel64_astra_ce_2.12`
	`  count: 1`

`linuxbox_force_clean_vms`	Позволяет принудительно удалять все использованные виртуальные машины перед клонированием новых. Значение по умолчанию: `False`.

13. Если на устройстве, где будет развернут сервис Dr.Web vxCube, используется продукт антивирусной защиты, исключите из проверки этим продуктом указанные ниже каталоги:

```
/proc
/var/lib/docker/volumes/drweb-service_storage-path
/var/lib/evparser_vxcube/in/
/var/lib/evparser_vxcube/out/
/var/lib/hyperbox/
/var/lib/storage/
/var/lib/storage/linuxbox-workdir/
/var/lib/storage/logs/
/var/lib/virtualenvs/hyperbox/lib
/tmp/pip-install*
/tmp/tmp*
/tmp/vxapi_hbarc_*
/tmp/vxapi_scan_*
/tmp/vxapi_sample_*
/opt/vxcube/tmp/vxcube_*
```

```
/var/lib/docker
/srv/vxcube/
```

Далее в качестве примера описывается исключение каталогов для антивируса Dr.Web Desktop Security Suite (Linux). Чтобы узнать, как исключить каталоги из проверки в продуктах других производителей, обращайтесь к соответствующим руководствам пользователя.

Чтобы исключить каталоги из проверки антивирусом Dr.Web Desktop Security Suite (Linux)

a) Запустите терминал.

b) Откройте конфигурационный файл антивируса с правами sudo в текстовом редакторе nano, выполнив следующую команду:

```
sudo nano /etc/opt/drweb.com/drweb.ini
```

Примечание. Вы можете использовать любой другой текстовый редактор. В этом случае потребуется изменить команду соответствующим образом.

c) В раздел `[LinuxSpider]` добавьте следующие строки:

```
LinuxSpider.ExcludedPath = /proc
LinuxSpider.ExcludedPath = /var/lib/docker/volumes/drweb-service_storage-path
LinuxSpider.ExcludedPath = /var/lib/evparser_vxcube/in/
LinuxSpider.ExcludedPath = /var/lib/evparser_vxcube/out/
LinuxSpider.ExcludedPath = /var/lib/hyperbox/
LinuxSpider.ExcludedPath = /var/lib/storage/
LinuxSpider.ExcludedPath = /var/lib/storage/linuxbox-workdir/
LinuxSpider.ExcludedPath = /var/lib/storage/logs/
LinuxSpider.ExcludedPath = /var/lib/virtualenvs/hyperbox/lib
LinuxSpider.ExcludedPath = /tmp/pip-install*
LinuxSpider.ExcludedPath = /tmp/tmp*
LinuxSpider.ExcludedPath = /tmp/vxapi_hbarc_*
LinuxSpider.ExcludedPath = /tmp/vxapi_scan_*
LinuxSpider.ExcludedPath = /tmp/vxapi_sample_*
LinuxSpider.ExcludedPath = /opt/vxcube/tmp/vxcube_*
LinuxSpider.ExcludedPath = /var/lib/docker
LinuxSpider.ExcludedPath = /srv/vxcube/
```

d) Сохраните конфигурационный файл.

e) Примените измененные настройки, выполнив следующую команду:

```
sudo drweb-ctl reload
```

Вы также можете добавить каталоги в исключения Dr.Web Desktop Security Suite (Linux), используя графический режим. Инструкции для этого режима приведены в руководстве пользователя продукта.

3.2. Установка на локальный сервер

Для установки всех компонентов Dr.Web vxCube на локальный сервер выполните команду:

```
$ make install
```

После выполнения команды будет запрошен пароль текущего пользователя посредством `BECOME password:`. Введите текущий пароль пользователя, чтобы Ansible установил компоненты из `inventory-local.yml` на сервер.

Производительность конфигурации может меняться в зависимости от технических характеристик устройства. Это стоит учитывать при формировании настроек в файле `vars-user.yml`.

3.3. Установка на удаленный сервер или группу серверов

Чтобы установить Dr.Web vxCube на один сервер или несколько серверов с одинаковой конфигурацией (количеством ядер, дисков и пр.)

1. В файле `inventory.yml` укажите нужные серверы для каждого компонента. Несколько серверов можно задать только в переменных `hyperbox_hosts`, `hyperbox_api_host`, `evparser_hosts`, `drweb_srv_hosts`, `dimas_hosts`, `linuxbox_hosts` и `yara_hosts`.
   Соответствующие этим переменным компоненты несут основную нагрузку при анализе файлов и позволяют горизонтально масштабировать Dr.Web vxCube для обработки больших объемов входящих файлов.

   Чтобы избежать зависания веб-интерфейса, рекомендуем устанавливать `vxcube_web_host` и анализаторы (`hyperbox_hosts`, `hyperbox_api_host`, `evparser_hosts`, `drweb_srv_hosts`, `dimas_hosts`, `linuxbox_hosts`, `yara_hosts`) каждый на свой узел.

2. Для установки на несколько серверов укажите в переменной `hyperbox_hosts` файла `inventory.yml` используемые диски, например:

```
hyperbox_hosts:
      hosts:
```

```
        192.168.1.10:
          hyperbox_ssds: [ "sda" ]
        192.168.1.11:
          hyperbox_ssds: [ "sda", "sdb", "sdc", "sdd"]
```

3. Для доступа к серверам укажите в файле `inventory.yml` имя пользователя и расположение приватного ключа (в переменных `ansible_user` и `ansible_ssh_private_key_file` соответственно).
   Этот пользователь должен иметь возможность запуска команд с привилегиями суперпользователя без запроса пароля. Чтобы создать такого пользователя на нескольких серверах, можно воспользоваться командой:

```
$ make prepare
```

   После ее выполнения пользователь будет создан на всех серверах, указанных в `inventory.yml`, а приватный ключ авторизации сохранится в указанном файле (по умолчанию — `credentials/ssh/id_rsa`).

4. Для запуска установки Dr.Web vxCube с помощью сформированного файла `inventory.yml` используйте команду:

```
$ make deploy
```

Чтобы установить Dr.Web vxCube на несколько серверов с различной конфигурацией (количеством ядер, дисков и пр.)

1. В файле `inventory.yml` укажите нужные серверы для каждого компонента. Несколько серверов можно задать только в переменных `hyperbox_hosts`, `hyperbox_api_host`, `evparser_hosts`, `drweb_srv_hosts`, `dimas_hosts`, `linuxbox_hosts` и `yara_hosts`.
   Соответствующие этим переменным компоненты несут основную нагрузку при анализе файлов и позволяют горизонтально масштабировать Dr.Web vxCube для обработки больших объемов входящих файлов.

   Чтобы избежать зависания веб-интерфейса, рекомендуем устанавливать `vxcube_web_host` и анализаторы (`hyperbox_hosts`, `hyperbox_api_host`, `evparser_hosts`, `drweb_srv_hosts`, `dimas_hosts`, `yara_hosts`) каждый на свой узел.

2. В переменной `hyperbox_hosts` файла `inventory.yml` укажите используемые диски, например:

```
hyperbox_hosts:
      hosts:
        192.168.1.10:
```

```
          hyperbox_ssds: [ "sda" ]
        192.168.1.11:
          hyperbox_ssds: [ "sda", "sdb", "sdc", "sdd"]
```

3. Для доступа к серверам укажите в файле `inventory.yml` имя пользователя и расположение приватного ключа (в переменных `ansible_user` и `ansible_ssh_private_key_file` соответственно).
   Этот пользователь должен иметь возможность запуска команд с привилегиями суперпользователя без запроса пароля. Чтобы создать такого пользователя на нескольких серверах, можно воспользоваться командой:

```
$ make prepare
```

   После ее выполнения пользователь будет создан на всех серверах, указанных в `inventory.yml`, а приватный ключ авторизации сохранится в указанном файле (по умолчанию — `credentials/ssh/id_rsa`).

4. Для каждого узла в каталоге `confs` разместите отдельный сертификат openvpn (`.crt`, `.key`), например: `192.168.1.10.crt`, `192.168.1.20.crt`, `192.168.2.10.key`, `192.168.2.20.key`.
5. В конфигурационном файле `vars-default.yml` переопределите значения переменных `openvpn_client_crt` и `openvpn_client_key`, например:

```
openvpn_client_crt: "{{ lookup('file', 'confs/
{{ inventory_hostname }}.crt') }}"
openvpn_client_key: "{{ lookup('file', 'confs/
{{ inventory_hostname }}.key') }}"
```

6. В корне раздела установочного архива создайте каталог `host_vars` и для каждого сервера создайте свой yml-файл с настройками для развертывания, например: `192.168.1.10.yml` и `192.168.2.20.yml`.
7. Заполните yml-файлы с настройками:
   a. Укажите пользователя Ansible и его пароль, например:

```
ansible_user: test_ansible_user
ansible_ssh_pass: test_ansible_user_pass
ansible_become_password: test_ansible_user_pass
```

   b. Укажите раздел, куда будут клонироваться образы виртуальных машин, например:

```
hyperbox_ssds: [ "sda" ]
```

   c. Укажите конфигурации виртуальных машин, например:

```
hyperbox_images:
  - vm_type: 6.1.7601.17514_x86
    code: Win7x86
```

```
    count: 2
    clone_threads: 2
    params:
      memory: 2112
      cores: 2

  - vm_type: 6.1.7601.17514_x64
    code: Win7x64
    count: 2
    clone_threads: 2
    params:
      memory: 2112
      cores: 2

linuxbox_images:
  - vm_type: intel64_astra_se_1.7.2
    code: intel64_astra_se_1.7.2
    count: 1

  - vm_type: intel64_astra_ce_2.12
    code: intel64_astra_ce_2.12
    count: 1

dimas_images:
  - vm_type: Android7.1
    code: Android7.1
    count: 3
    memory: 4072
    cores: 2
    clone_threads: 3
```

Переменные, указанные в `vars-default.yml`, имеют более высокий приоритет при развертывании. Чтобы переопределить их как переменные, которые вы внесли в yml-файлы каталога `host_vars`, закомментируйте их в `vars-default.yml`.

Например, если мы создали `host_vars/192.168.1.10.yml` и там переопределили переменную `hyperbox_ssds: [ "sda" ]`, то переменную `hyperbox_ssds` в `vars-default.yml` нужно закомментировать.

8. Для запуска установки Dr.Web vxCube с помощью сформированного файла `inventory.yml` используйте команду:

```
$ make deploy
```

Более подробные сведения о файле `inventory.yml` и работе с ним приведены в документации Ansible.

3.4. Как обновить настройки после установки

Если вам нужно обновить настройки после установки Dr.Web vxCube, перезапустите инсталлятор с опцией `hyperbox_hbsetup: false`. Это позволит не разворачивать виртуальные машины повторно, что значительно ускорит установку.

Вы также можете изменить часть настроек вручную на сервере.

Обновление SSL-сертификата

Чтобы обновить SSL-сертификат

1. Добавьте новые файлы по следующим путям:
   - `/etc/nginx/ssl/vxcube.crt`,
   - `/etc/nginx/ssl/vxcube.key`.
2. Перезагрузите веб-сервер:

```
sudo systemctl reload nginx
```

Обновление настроек VPN-агента

Чтобы изменить IP-адрес сервера

1. Откройте файл `/etc/openvpn/client.conf`.
2. Укажите новый адрес в параметре `remote "Новый IP-адрес" 1194`.

Чтобы обновить сертификаты доступа к VPN-серверу

1. Добавьте новые файлы по следующим путям:
   - `/etc/openvpn/client.crt`,
   - `/etc/openvpn/client.key`,
   - `/etc/openvpn/ca.crt`.

Чтобы изменить IP-адрес VPN-шлюза

1. Откройте файл `/etc/vbox/config.json`.
2. Укажите новый IP-адрес VPN-шлюза: `"vpn_gateway": "IP-адрес VPN-шлюза"`.

После внесения всех изменений перезагрузите сервисы OpenVPN и Sandbox:

```
sudo systemctl restart openvpn vboxsvc vboxapi
```

Обновление настроек сетевого интерфейса

Настройки сетевого интерфейса по умолчанию передаются через протокол DHCP.

Чтобы изменить настройки

1. Откройте файл `/etc/netplan/01-netcfg.yaml`.
2. Задайте новые настройки.
3. Выполните `netplan apply`, чтобы применить изменения.
4. Перезагрузите сервер.

3.5. Список служб Dr.Web vxCube

В таблице ниже представлен список служб, устанавливаемых инсталлятором vxCube (помимо служб, входящих в состав ОС Astra Linux 1.7.3), и пути к соответствующим журналам.

Служба	Расположение журнала/команда для просмотра журнала	Описание
Службы инфраструктуры		
nginx.service	`/var/log/nginx`	Высокопроизводительный веб-сервер и обратный прокси-сервер
openvpn.service	`/var/log/openvpn`	Служба OpenVPN
openvpn@client.service	`sudo journalctl -u openvpn@client.service`	Соединение OpenVPN с клиентом
proftpd.service	`/var/log/proftpd`	Запуск управляющей программы ProFTPD
containerd.service	`sudo journalctl -u containerd.service`	Среда выполнения контейнера containerd
docker.service	`sudo journalctl -u docker.service`	Docker Application Container Engine
rabbitmq-server.service	`/var/log/rabbitmq/`	Брокер сообщений RabbitMQ
Базовые службы виртуализации		
vboxdrv.service	`sudo journalctl -u vboxdrv.service`	Модуль ядра Linux для VirtualBox
vboxnet.service	`sudo journalctl -u vboxnet.service`	VirtualBox Network Service

Служба	Расположение журнала/команда для просмотра журнала	Описание
vboxsvc.service	`sudo journalctl -u vboxsvc.service`	Служба VirtualBox
vboxapi.service	`sudo journalctl -u vboxapi.service`	Служба VirtualBox API
vboxautostart-service.service	`sudo journalctl -u vboxautostart-service.service`	Служба автозапуска VirtualBox
vboxballoonctrl-service.service	`sudo journalctl -u vboxballoonctrl-service.service`	Сторожевой таймер VirtualBox
vboxweb-service.service	`sudo journalctl -u vboxweb-service.service`	Веб-служба VirtualBox API
Службы виртуализации Windows		
hyperbox_<*>_vxcube.service, где <*> — имя образа Windows, например hyperbox_win10x64_1903_vxcube.service	`/var/log/hyperbox/`	Celery Worker для hyperbox_<*>_vxcube 1,3, где <*> — имя поставляемого с инсталлятором образа Windows, например `win7x86`, `win10x64_1903`, `win7x64`, `winxpx86` (может быть несколько подобных служб)
hbcheck.service	`sudo journalctl -u hbcheck.service`	Проверка Hyperbox
Службы виртуализации Android		
dimas_<*>vxcube.service, где <*> — имя образа Android, например dimas_android4.3_vxcube.service	`/var/log/dimas`	Celery Worker Dimas для dimas_<*>_vxcube 1,1, где <*> — имя поставляемого с инсталлятором образа Android, например `android4.3`, `android7.1` (может быть несколько подобных служб)
dimasnet.service	`sudo journalctl -u dimasnet.service`	Служба dimasnet vboxifs init oneshot service
vboxapi_android.service	`/var/log/dimas/vboxapi`	Служба API Android VirtualBox
Системные, сетевые утилиты		

Служба	Расположение журнала/команда для просмотра журнала	Описание
binfmt-support.service	`sudo journalctl -u binfmt-support.service`	Поддержка дополнительных исполняемых бинарных форматов
loadcpufreq.service	`sudo journalctl -u loadcpufreq.service`	Загрузка модулей ядра для масштабирования CPUFreq
cpufrequtils.service	`sudo journalctl -u cpufrequtils.service`	Задает параметры CPUFreq ядра
netfilter-persistent.service	`sudo journalctl -u netfilter-persistent.service`	netfilter persistent configuration
isc-dhcp-server.service	`sudo journalctl -u isc-dhcp-server.service`	ISC-DHCP-сервер с IPv4
Служба анализатора		
evparser.service	`/var/log/evparser`	Служба EvParser
Служба vxCube Flow API		
vxcube-flow-api.service	`/var/log/vxcube-flow-api`	HyperboxAPI
Службы vxCube, выполняемые в Docker-контейнерах		
vxcube-web	`/var/log/vxcube/testing`	Веб-интерфейс vxCube Файл docker-compose: `/var/lib/vxcube/active/docker-compose.yml`
vxcube-redis	В директории, где расположены файлы docker-compose: `sudo docker-compose logs`	Файл docker-compose: `/var/lib/vxcube/active/docker-compose.yml`
vxcube-postgres	В директории, где расположены файлы docker-compose: `sudo docker-compose logs`	Файл docker-compose: `/var/lib/vxcube/active/docker-compose.yml`
yara-service	`/var/log/yara_service`	Файл docker-compose: `/etc/yara_service/docker-compose.yml`
drweb-service_drweb-srv_1	`/var/log/drweb`	Файл docker-compose: `/etc/drweb-service/docker-compose.yml`

Служба	Расположение журнала/команда для просмотра журнала	Описание
drweb-service_drweb-se_1	В директории, где расположены файлы docker-compose: `sudo docker-compose logs`	Файл docker-compose: `/etc/drweb-service/docker-compose.yml`

Для просмотра журнала в реальном времени используйте команду `tail -f <путь к файлу лога>`.

Если вы используете для просмотра журнала команды `jourtnalctl` и `docker-compose logs`, для просмотра данных в реальном времени укажите ключ `-f`.

Для сбора данных о событиях для службы технической поддержки можно также воспользоваться следующим скриптом:

```
if [ "$EUID" -ne 0 ]
  then echo "Please run as root"
  exit
fi
rm -rf support.tar.gz support.tar
set -x
ifconfig > ifconfig.log
journalctl -b > journal.log
tar -P -cf
support.tar /var/log/drweb /var/log/evparser /var/log/vxcube/testing /var/log/
nginx /var/log/hyperbox /var/log/openvpn /var/log/proftpd ifconfig.log
journal.log /var/lib/hyperbox/hbsetup.log
find "/var/lib/hyperbox/VirtualBox VMs/" -type d -name "Logs" -exec tar -P -
rvf support.tar {} \;
rm -rf ifconfig.log
rm -rf journal.log
gzip support.tar
set +x
```

Для перезапуска служб, выполняемых в Docker-контейнерах

1. Перейдите в директорию, где расположены соответствующие файлы docker-compose.
2. Выполните команды перезапуска.

Для `/var/lib/vxcube/active/docker-compose.yml`:

```
cd /var/lib/vxcube/active/

sudo docker-compose down

sudo docker-compose up -d
```

Для `/etc/yara_service/docker-compose.yml`:

```
cd /etc/yara-service

sudo docker-compose down

sudo docker-compose up -d
```

Для `/etc/drweb-service/docker-compose.yml`:

```
cd /etc/drweb-service

sudo docker-compose down

sudo docker-compose up -d
```

Для мониторинга состояния компонентов vxCube опционально можно установить агент Zabbix.

4. Как обновить Dr.Web vxCube

При выходе новой версии дистрибутива Dr.Web vxCube и образов виртуальных машин вы можете обновить установленный у вас сервис.

Для этого выполните указанные ниже действия:

1. Скачайте дистрибутив и образы. Распакуйте скачанные архивы. Дистрибутив нужно распаковывать в каталог, из которого будет производиться установка.

   Не рекомендуем распаковывать архив с дистрибутивом в тот же каталог, откуда производилась установка предыдущей версии сервиса. Выберите другой каталог.

2. Поместите лицензионный ключ Dr.Web vxCube в каталог `confs/vxcube.key` распакованного дистрибутива.
3. Поместите файлы дистрибутива и образов в следующие каталоги:
   - SSL-сертификаты для подключения к VPN-серверам, указанным в файле `vars-user.yml` (переменная `openvpn_client_servers`), в каталоги `confs/openvpn.crt`, `confs/openvpn.key` и `confs/openvpn_ca.crt`;

     SSL-сертификаты не являются частью продукта Dr.Web vxCube; их необходимо приобрести отдельно.

   - полученные образы виртуальных машин на базе Windows с расширением `.tar.gz` и все служебные файлы `.tar.gz.ver` и `.tar.gz.hash` в каталог `vm-images-win`;
   - полученные образы виртуальных машин на базе Android с расширением `.vdi` и все служебные файлы `.vdi.hash` и `.vdi.ver` в каталог `vm-images-andr`;
   - полученные образы виртуальных машин на базе Linux с расширением `.tar.gz` и все служебные файлы `.tar.gz.hash` и `.tar.gz.ver` в каталог `vm-images-linux`.

     Вы также можете поместить образы виртуальных машин в произвольные каталоги. В этом случае вам нужно указать эти каталоги в файле `vars-default.yml` (в переменных `hyperbox_images_repo`, `dimas_images_repo` и `linuxbox_images_repo`).

4. Если вы хотите, чтобы веб-интерфейс Dr.Web vxCube поддерживал работу по протоколу HTTPS, выполните следующие действия:
   - создайте файлы `confs/web_ssl.crt`, `confs/web_ssl.key`;
   - создайте файл `confs/web_dhparam.pem` (если пропустили шаг 5);
   - в файле `vars-default.yml` раскомментируйте переменные `vxcube_web_ssl_cert`, `vxcube_web_ssl_privkey` и `vxcube_web_dhparam`.
5. Обновите ОС Astra Linux до версии 1.7.6. В `/etc/apt/sources.list` должны быть указаны следующие репозитории:

```
# Расширенный репозиторий

deb https://dl.astralinux.ru/astra/stable/1.7_x86-64/repository-
extended/ 1.7_x86-64 main contrib non-free

# Репозиторий Astra 1.7.6

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-
64/1.7.6/repository-main/ 1.7_x86-64 main contrib non-free

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-
64/1.7.6/repository-update/ 1.7_x86-64 main contrib non-free

deb http://dl.astralinux.ru/astra/frozen/1.7_x86-
64/1.7.6/repository-base/ 1.7_x86-64 main contrib non-free
```

6. После обновления репозиториев выполните команды:

```
sudo apt update
sudo apt upgrade
```

7. Отредактируйте настройки установки. Для этого в файле `vars-user.yml` задайте значения всех переменных. Их подробное описание вы найдете в самом файле, а также в таблице Описание переменных в файле vars-user.yml.

   Дополнительно вы также можете указать или изменить значения переменных в файле `vars-default.yml`. Их подробное описание вы найдете в самом файле, а также в таблице Описание переменных в файле vars-default.yml.

8. Скопируйте файлы `hyperbox_api_rq_password`, `vxcube_ftp_pass` и `vxcube_web_db_pass` из каталога `/credentials` прошлой установки продукта в каталог `/credentials` новой установки.

9. Выполните команды отключения старой службы EvParser:

```
sudo systemctl stop evparser.service
sudo systemctl disable evparser.service
```

Выполнив все указанные выше шаги, вы можете перейти к установке продукта на локальный сервер или на удаленный сервер.

5. Как войти в Dr.Web vxCube и выйти из учетной записи

Вход в Dr.Web vxCube

Перед началом работы с Dr.Web vxCube убедитесь, что ваш компьютер соответствует системным требованиям.

Чтобы войти в Dr.Web vxCube

1. Перейдите по адресу https://<*IP сервера*> или https://<*доменное имя сервера*>.
2. Введите свой логин (адрес электронной почты). Он указан в файле `vars-user.yml` в переменной `vxcube_web_superuser_email`.
3. Введите пароль. Он указан в файле `vars-user.yml` в переменной `vxcube_web_superuser_pass`. Если значение этой переменной в файле не задано, значит пароль был сгенерирован автоматически и его можно найти в файле `vxcube_web_superuser_pass` каталога `credentials`.

При первом входе в Dr.Web vxCube требуется принять Лицензионное соглашение.

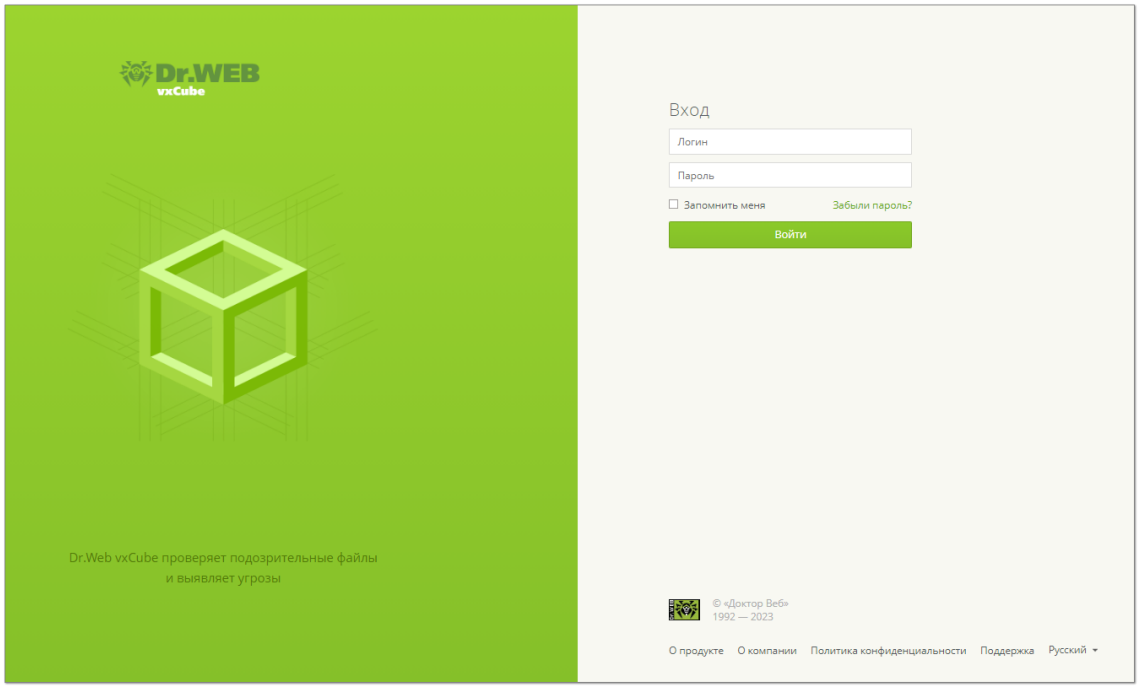


Рисунок 1. Страница входа в Dr.Web vxCube

Выход из учетной записи Dr.Web vxCube

Чтобы выйти из учетной записи Dr.Web vxCube, в правом верхнем углу главной страницы нажмите **Профиль** > **Выход**.

6. Настройки

Вы можете менять язык интерфейса Dr.Web vxCube (в настоящее время поддерживаются русский и английский языки), указывать настройки анализа файлов, которые будут использоваться по умолчанию, а также управлять своими API-ключами и паролем.

6.1. Как изменить язык интерфейса

Сервис Dr.Web vxCube доступен на русском и английском языках. По умолчанию язык интерфейса Dr.Web vxCube совпадает с языком интерфейса браузера, в котором используется сервис.

Чтобы изменить язык интерфейса

1. Пролистайте страницу сервиса вниз.
2. Нажмите поле выбора языка в нижней части страницы.
3. Выберите необходимый язык в открывшемся списке.

6.2. Настройки анализа по умолчанию

Вы можете задать следующие настройки анализа по умолчанию: время выполнения файла в виртуальной машине, версии ОС, для которых будет проводиться анализ, пароль для архива отчета (если пароль не задан, архив высылается без пароля).

В поле **Пароли для архивов исходных файлов** можно добавить пароли, которые будут использованы при попытке анализа защищенного паролем архива.

Если для архива с отчетом не задан пароль, антивирус на локальном устройстве может его проанализировать и в некоторых случаях определить как угрозу. Например, если в отчете содержатся дампы alloc-функций.

Чтобы задать настройки анализа по умолчанию

1. В правом верхнем углу главной страницы Dr.Web vxCube нажмите **Профиль** > **Настройки**.
2. Выберите слева вкладку **Анализ**.
3. Укажите нужные настройки по умолчанию для проведения анализа файлов.

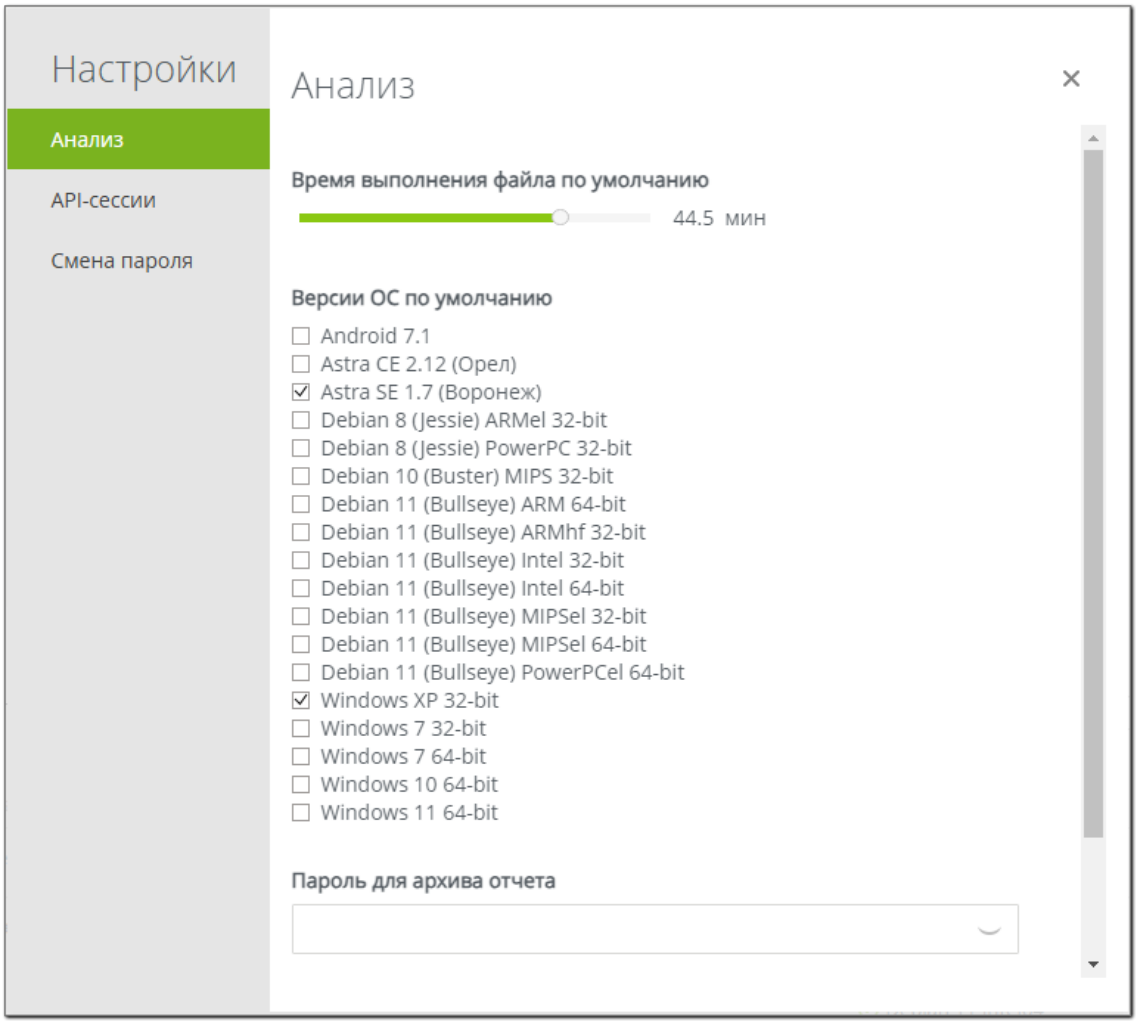


Рисунок 2. Настройки

6.3. Управление паролем

Вы можете сбросить пароль, если забыли его. Кроме того, в целях защиты своей учетной записи вы можете сменить пароль.

6.3.1. Как сменить пароль

Чтобы сменить пароль

1. В правом верхнем углу главной страницы Dr.Web vxCube нажмите **Профиль** > **Настройки**.
2. Выберите слева вкладку **Смена пароля**.
3. Укажите действующий пароль, затем дважды введите новый пароль и нажмите **Сохранить**.

6.3.2. Как сбросить пароль

Поскольку Dr.Web vxCube использует электронную почту для восстановления паролей, на его сервере должен быть настроен SMTP-сервер. Администратор также может указать другой SMTP-сервер в переменной `vxcube_web_mail_server` в файле `vars-default.yml`.

Чтобы сбросить существующий пароль

1. На странице входа в Dr.Web vxCube нажмите **Забыли пароль?**.

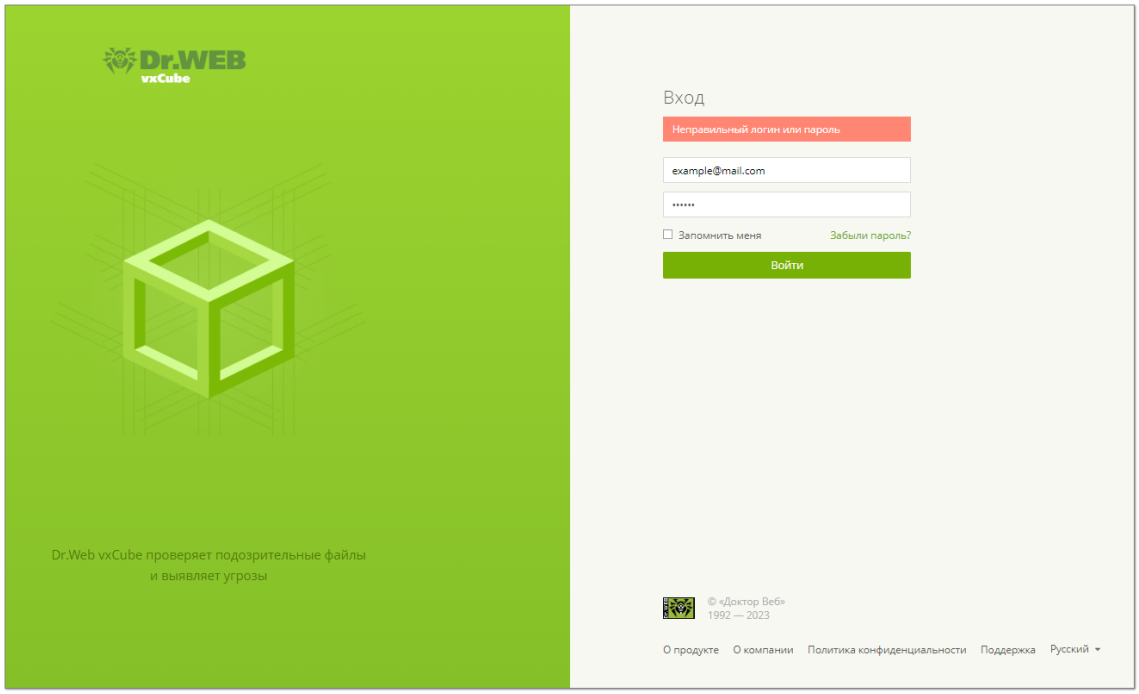


Рисунок 3. Ошибка при входе в Dr.Web vxCube

2. На странице **Сброс пароля** укажите адрес электронной почты, который вы использовали при регистрации.
3. Установите флажок **Я не робот**. Это необходимо только в случае, если при установке Dr.Web vxCube была включена проверка пользователя.
4. Нажмите **Отправить**.

   На указанный адрес электронной почты будет отправлено письмо со ссылкой для сброса пароля. Если вы не получили письмо в течение 10 минут, проверьте папку Спам или свяжитесь с администратором сервиса.

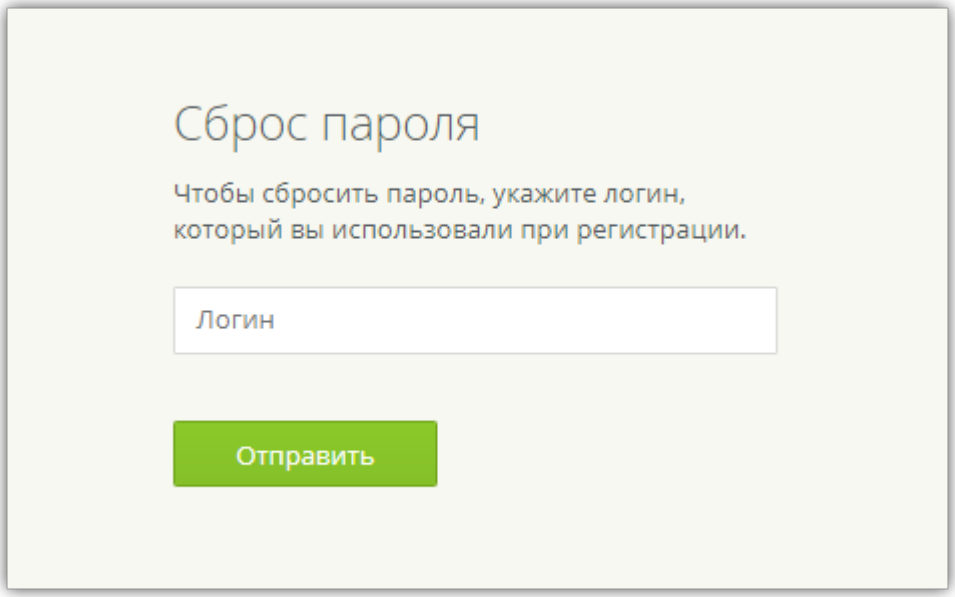


Рисунок 4. Отправка запроса на сброс пароля

5. Откройте полученное письмо.
6. Чтобы сбросить пароль, перейдите по ссылке в теле письма.

   Вы будете перенаправлены на страницу Dr.Web vxCube.

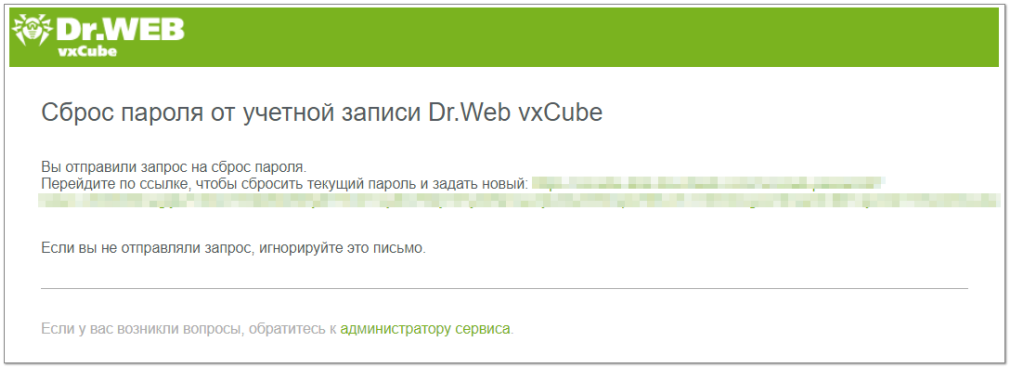


Рисунок 5. Подтверждение сброса пароля

7. Укажите новый пароль и подтвердите его.
8. Нажмите **Создать**.

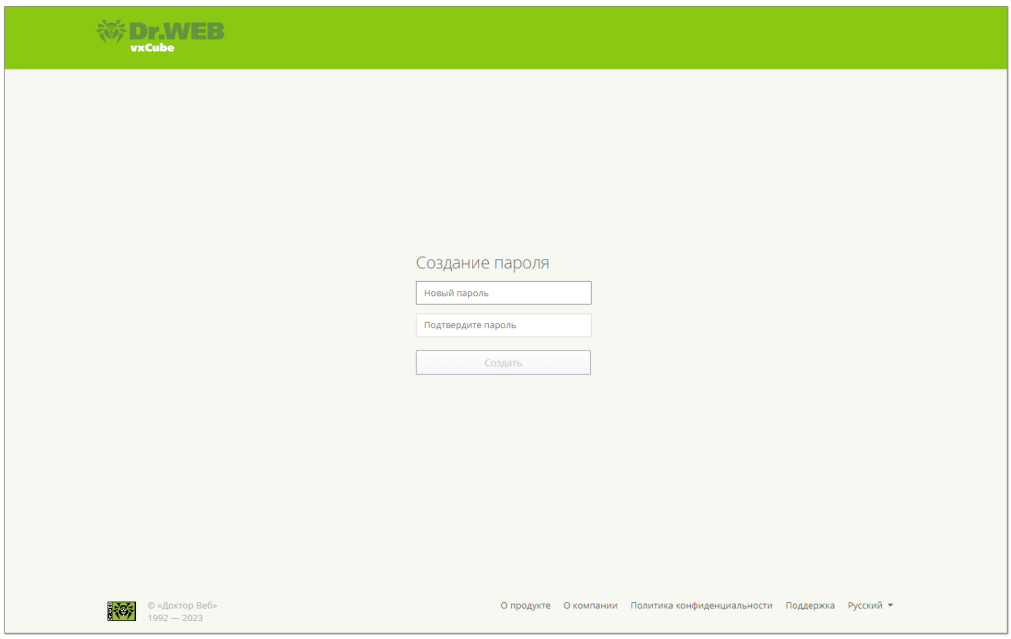


Рисунок 6. Создание пароля

7. Лицензирование

Чтобы использовать Dr.Web vxCube, вам нужно приобрести лицензию на продукт. При покупке лицензии вы сможете выбрать подходящие вам опции:

- платформы, на которых будут анализироваться файлы (Windows, Android, Linux);
- срок действия лицензии.

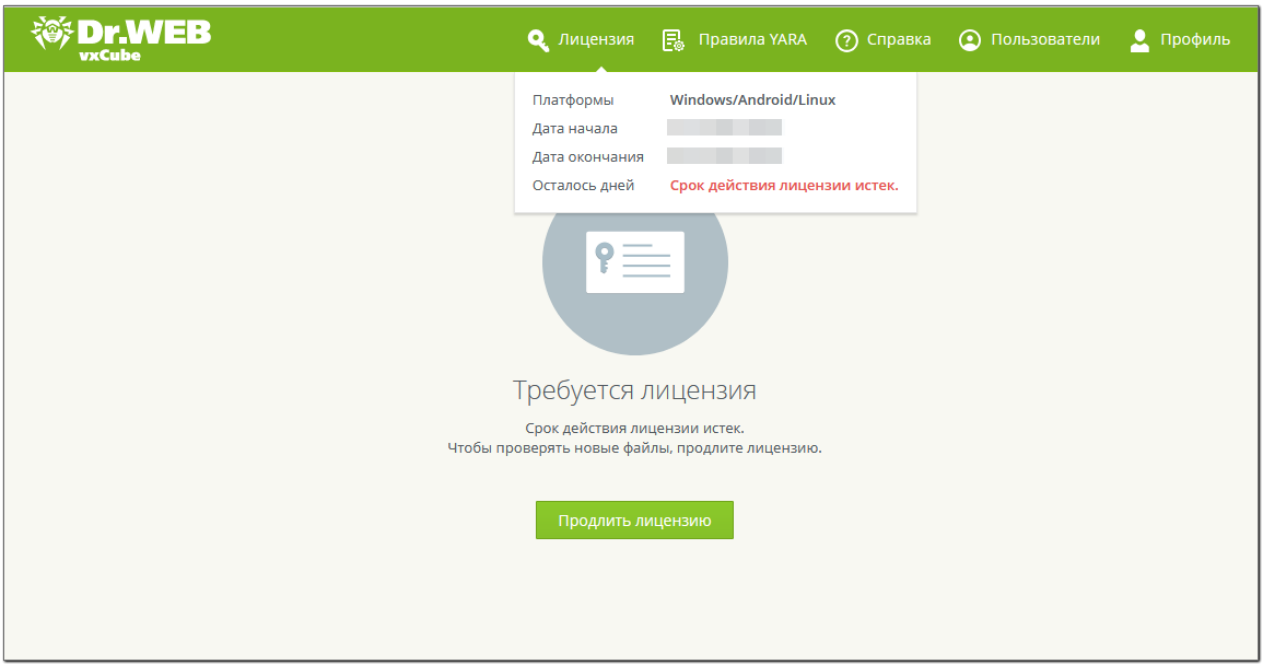
После того как срок действия лицензии истечет, вы больше не сможете загружать в сервис файлы и проверять их. При этом вы по-прежнему сможете:

- заходить на страницу сервиса;
- просматривать информацию о проверенных ранее файлах;
- скачивать архивы с отчетами о проверке.

Чтобы снова пользоваться всеми функциями Dr.Web vxCube, продлите лицензию.

Продление лицензии

Когда срок действия лицензии истечет, на главной странице Dr.Web vxCube появится соответствующее сообщение.



Чтобы снова получить доступ ко всем возможностям сервиса Dr.Web vxCube, нажмите **Продлить лицензию**. Откроется страница покупки лицензии на сайте «Доктор Веб», где вы сможете заказать новый лицензионный файл ключа. Получив его, вручную обновите лицензию, как описано ниже.

Чтобы вручную обновить лицензию vxCube

1. Измените имя полученного нового ключа `drweb32.key` на `vxcube.key` и сохраните его в каталоге `/etc/vbox/vxcube.key`, заменив уже имеющийся там файл.
2. Измените файл конфигурации лицензии для службы vxcube-web:
   a) В файле `/opt/vxcube/config/default.yml` замените значения полей `activation_date` и `expire_date` значениями полей `Created` и `Expires` из файла ключа.

      Внимание! Изменяйте файл не внутри Docker-контейнера, а на хосте, где запущен Docker-контейнер, поскольку файл проброшен в том на хосте.

   b) Перезапустите службу vxcube-web с помощью команд:

```
cd /var/lib/vxcube/active

sudo docker-compose down

sudo docker-compose up -d
```

Чтобы вручную обновить лицензию сервиса сканирования файлов (drweb-service)

1. Сохраните новый файл лицензии в каталоге `/etc/drweb-service/drweb32.key` на хосте, где запущен Docker-контейнер (но не внутри Docker-контейнера, поскольку файл проброшен в том на хосте).
2. Перезапустите сервис с помощью команд:

```
cd /etc/drweb-service

sudo docker-compose down

sudo docker-compose up -d
```

Информация о текущей лицензии

Чтобы посмотреть информацию о параметрах своей лицензии, в верхней части главной страницы Dr.Web vxCube нажмите **Лицензия**. Откроется окно со следующей информацией о вашей лицензии:

- доступные платформы, на которых могут анализироваться файлы (Windows, Android, Linux);
- количество дней, оставшееся до истечения срока действия лицензии;
- дата и время начала срока действия лицензии;
- дата и время окончания срока действия лицензии.

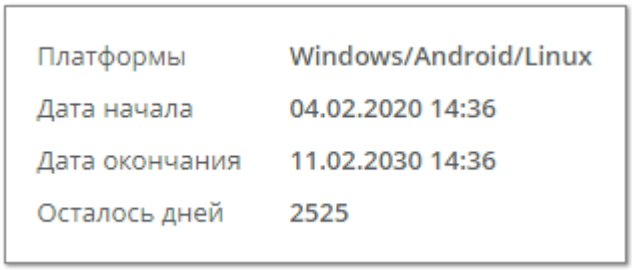


Рисунок 7. Информация о лицензии

8. Правила YARA

Используя правила YARA, вы можете находить и классифицировать вредоносные объекты: правило сработает, если выполнится заданное в нем условие. Таким условием может быть определенное содержимое, поведение или место обнаружения файла. Правила YARA могут включать в себя строки, логические выражения, подстановочные знаки, регулярные выражения, специальные операторы и множество других функций. Подробную информацию о создании правил YARA можно найти в официальной документации YARA.

Правила YARA, используемые в Dr.Web vxCube, имеют ряд особенностей:

- в раздел правил `meta` добавлено обязательное поле `maliciousness`, с помощью которого указывается тип вредоносности, выставляемый в отчете при срабатывании правила;
- с помощью специального модуля dr_sandbox можно создавать правила, которые будут срабатывать при обнаружении определенного поведения файла в виртуальной машине.

Все правила YARA в сервисе Dr.Web vxCube делятся на *системные* и *пользовательские*. Системные правила создаются разработчиками Dr.Web vxCube и по умолчанию используются при анализе файлов. Вы не можете изменять и удалять такие правила, но можете просматривать их код и выключать правила, которые вам не нужны. Пользовательские правила вы создаете сами. Их можно изменять, выключать и удалять.

8.1. Как создать правило YARA

Для всех правил YARA в сервисе Dr.Web vxCube используется следующий стандартный формат:

```
rule RuleName1 : TAG1 TAG2
{
    meta:
        maliciousness = "neutral"

    strings:
        $s = "SomeString"

    condition:
        $s
}
```

Правило всегда начинается с *ключевого слова* `rule`. За ним следует *имя правила*. Имя может состоять из букв латинского алфавита, цифр и подчеркиваний; оно не может начинаться с цифры. Затем после двоеточия указываются *теги* (необязательно). Если правило сработает, эти теги попадут в отчет. Тег, как и имя правила, не должен начинаться с цифры. Далее следует *тело правила*. Оно может состоять из трех разделов:

- В обязательном разделе `meta` указывается тип вредоносности (поле `maliciousness`), который будет выставлен для файла, если правило сработает. Возможные значения для поля `maliciousness`: `neutral`, `suspicious`, `malware`.
- В обязательном разделе `condition` задается условие, при выполнении которого правило сработает.
- В необязательном разделе `strings` указываются строки, используемые в правиле.

Чтобы создать правило YARA

1. В верхней части главной страницы Dr.Web vxCube нажмите **Правила YARA**.
2. Нажмите ⊕ **Добавить**. Откроется окно с шаблоном правила.
3. Заполните шаблон, указав нужные вам параметры правила.
4. Нажмите **Добавить**.

Рисунок 8. Создание правила YARA

8.2. Как управлять правилами YARA

Чтобы посмотреть все правила YARA, доступные для вашей учетной записи, в верхней части главной страницы Dr.Web vxCube нажмите **Правила YARA**. Откроется страница со списком правил. Для каждого правила в списке отображается следующая информация:

- Тип правила (значок для пользовательского и для системного правила).
- **Имя.** Имя правила.
- **Вредоносность.** Степень вредоносности, заданная в правиле.
- **Теги.** Теги, заданные в правиле.
- **Срабатывания.** Общее количество раз, которое сработало это правило.
- **Последнее срабатывание.** Дата последнего срабатывания. Если последний раз правило сработало в тот же день, когда вы просматриваете список, вместо даты указывается время.
- **Состояние.** Текущее состояние правила (включено или выключено).

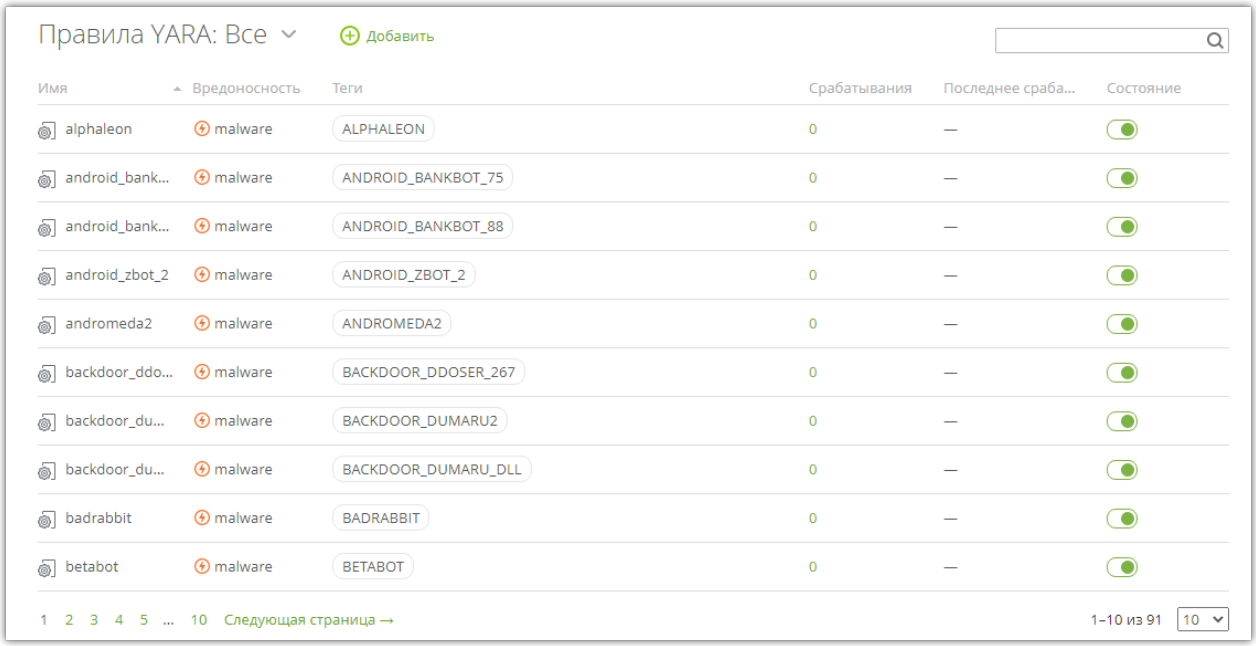


Рисунок 9. Список правил YARA

На странице со списком правил YARA вы можете:

- искать правила по имени и тегам;
- фильтровать правила по типу (системные, пользовательские);
- сортировать правила;
- просматривать информацию о срабатываниях правила (имя файла, при анализе которого сработало правило, дата срабатывания, ОС);
- просматривать код системных правил;
- изменять, удалять, включать и выключать правила.

Чтобы найти правило

- Введите имя правила или тег (целиком или частично) в поле поиска, расположенном над списком правил справа.

Чтобы отфильтровать правила по типу

- Справа от заголовка списка нажмите значок ⌄ и выберите нужный вариант фильтра: **Правила YARA: Все**, **Правила YARA: Системные** или **Правила YARA: Пользователь**.

Чтобы отсортировать правила

- Нажмите заголовок соответствующего столбца. Справа от заголовка столбца, по которому на данный момент сортируются правила, отображается значок ▲ или ▼. Чтобы изменить направление сортировки, еще раз нажмите заголовок.

Чтобы просмотреть информацию о срабатываниях правила

- Нажмите число в столбце **Срабатывания** для нужного вам правила. Откроется страница отчетов о срабатываниях для этого правила.

Чтобы просмотреть код системного правила

- Наведите курсор на строку с правилом и нажмите справа значок  .

Чтобы изменить правило

- Наведите курсор на строку с правилом и нажмите справа значок  .

Чтобы удалить правило

- Наведите курсор на строку с правилом и нажмите справа значок  .

Чтобы включить или выключить правило

- В строке с правилом переведите переключатель  в нужное положение.

Чтобы задать количество правил, отображаемых на одной странице

- Выберите нужное значение (10, 25, 50 или 100) в выпадающем меню справа под списком.

8.3. Отчеты о срабатываниях правила

Вы можете посмотреть информацию обо всех срабатываниях определенного правила YARA. Для этого:

1. В верхней части главной страницы Dr.Web vxCube нажмите **Правила YARA**.
2. Нажмите число в столбце **Срабатывания** для нужного вам правила.

Откроется список всех срабатываний этого правила. Для каждого срабатывания отображается следующая информация:

- **Имя файла.** Имя файла, при анализе которого произошло срабатывание.
- **Формат.** Формат файла, при анализе которого произошло срабатывание.
- **SHA1.** Хеш файла.
- **Дата.** Дата срабатывания правила.
- **ОС.** Список операционных систем, в рамках которых проводился анализ.

Из списка срабатываний вы можете перейти к отчету об анализе, в рамках которого произошло конкретное срабатывание:

- Чтобы перейти на главную страницу отчета, нажмите имя файла в строке срабатывания.
- Чтобы перейти на страницу отчета для определенной операционной системы, нажмите название этой операционной системы в строке срабатывания.

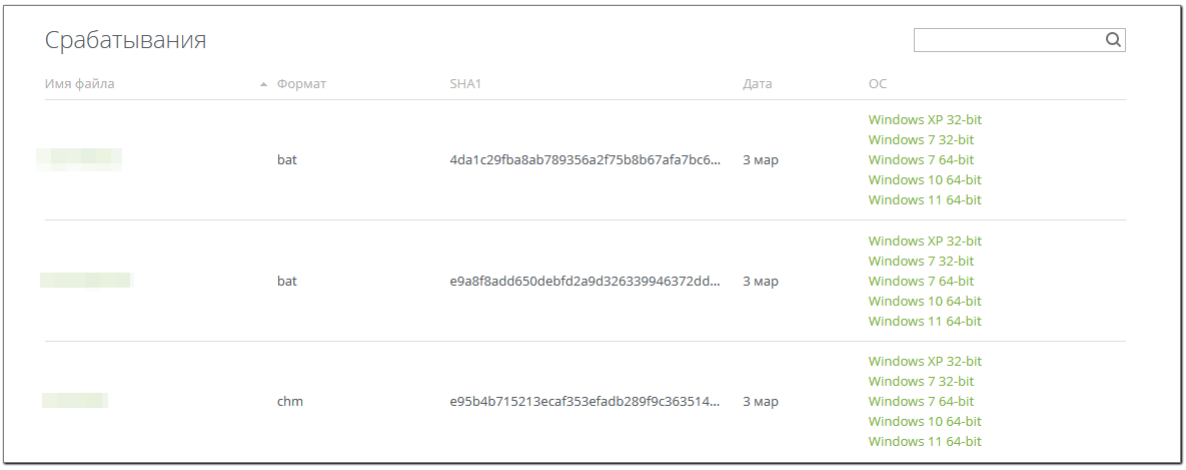


Рисунок 10. Отчеты о срабатываниях правила YARA

8.4. Модуль dr_sandbox

Модуль dr_sandbox — это эксклюзивный модуль YARA компании «Доктор Веб», с помощью которого вы можете создавать правила на основе следующей информации:

- поведение файла в виртуальной машине;
- тип создаваемых файлов (`src`, `dump`, `drop`, `alloc` и т. д.);
- сведения об обнаруженных угрозах;
- имя анализируемого файла.

Пример правила, в котором используется функция `connect_to` модуля dr_sandbox:

```
rule bad_file
{
    condition:
        dr_sandbox.descr_tech.network.connect_to(/http:\/\/someplace\.badsite\.com/)
}
```

Список всех функций модуля dr_sandbox с описанием и примерами приведен в Приложении Б. Функции модуля dr_sandbox.

9. Анализ файлов

Чтобы проанализировать файл

1. Убедитесь, что формат файла есть в списке поддерживаемых.
2. Выберите файл, который хотите проанализировать, и загрузите его в приложение.

   Если Dr.Web vxCube не сможет определить формат файла, вы сможете указать его вручную.
3. Укажите версию операционной системы, в которой хотите провести анализ, или версию приложения.

   Вы можете выбрать несколько версий ОС или приложений для запуска файла.
4. При желании задайте дополнительные настройки для анализа.
5. Нажмите **Анализировать**.

Также присутствует возможность анализа файлов через API.

Процесс анализа

После того как вы начнете анализ, запустится виртуальная машина или несколько машин в зависимости от того, сколько операционных систем или версий приложения вы выбрали, с предустановленным программным обеспечением.

Для выявления подозрительных действий на запущенной виртуальной машине поведение файла тщательно отслеживается. Все процессы, происходящие на гостевой операционной системе во время запуска файла, заносятся в Журнал API. Используя список правил, анализатор распределяет действия по категориям.

Анализатор Dr.Web vxCube работает на уровне *гипервизора* и не использует дополнительное программное обеспечение (например, специальные драйверы для перехвата функций) в гостевой операционной системе. Это не позволяет исследуемому файлу обнаруживать или снимать перехваты.

В процессе анализа виртуальные машины могут выходить в интернет через выделенный прокси-сервер. Это позволяет анализировать поведение вируса в полном объеме, особенно если его работа напрямую зависит от загрузки данных из сети.

Журнал событий также ведется на уровне гипервизора, а не виртуальной машины, что делает обнаружение анализатора невозможным.

Вы можете подключиться к виртуальной машине с помощью VNC-клиента (Virtual Network Computing) и влиять на процесс анализа, однако это возможно только во время работы виртуальной машины.

По результатам анализа вы получите отчет, а также сможете просмотреть журнал ранее проанализированных файлов.

Иногда анализ одного и того же файла может приводить к различным результатам, если поведение файла зависит от внешних условий, например, текущего времени или доступности удаленных ресурсов.

Кроме того, результаты анализа с использованием VNC могут отличаться от результатов анализа без VNC, если анализируемый файл использует способ инжекта, неизвестный Dr.Web vxCube, или управление передается процессам опосредованно.

9.1. Поддерживаемые форматы файлов

Dr.Web vxCube поддерживает анализ следующих форматов файлов:

Тип файлов	Формат файлов
Исполняемые файлы Windows	CPL, DLL, EXE, MSI, NATIVE APP, SYS
Пакеты Android	APK
Файлы Microsoft Office	ACCDB, DOC, DOCM, DOCX, DOTM, DOTX, IQY, MHT, ODP, ODS, ODT, POTM, POTX, PPA, PPAM, PPSM, PPSX, PPT, PPTM, PPTX, PUB, RTF, SLDM, SLDX, SLK, THMX, WPS, XLAM, XLL, XLS, XLSB, XLSM, XLSX, XML, XTLM, XTLX
Файлы Acrobat Reader	PDF
Исполняемые файлы Java	CLASS, JAR
Файлы сценарных языков	BAT, JS, JSE, PL, PS1, PY, SCT, SH, VBE, VBS, WSF, XSL
Исполняемые файлы *nix	ELF
Другие	7Z, ACE, ARJ, BZ2, CAB, CHM, DOCKER, EML, GZ, HTA, LNK, MOF, RAR, TAR, XZ, ZIP

- Архивы (файлы с расширениями 7Z, ACE, ARJ, BZ2, CAB, EML, GZ, RAR, TAR, XZ и ZIP) можно отправить на анализ только через API.
- Архив не анализируется целиком. Для каждого файла архива потребуется отправить отдельный API-запрос. Пример

Размер загружаемого файла не должен превышать 1000 МБ.

Особенности обработки файлов

В зависимости от формата каждого файла в Dr.Web vxCube используются разные механизмы его обработки и способы запуска.

При выборе файлов Microsoft Office, Acrobat Reader и Java выбор операционной системы заменяется на выбор версии соответствующего приложения. Например, для PDF-файла появится выбор между 10.1, 11.0 и 15.10 версиями Acrobat Reader.

Форматы файлов и способы их запуска

Формат файла	Способ запуска
EXE	*%sample%*
DLL	regsvr32 /s *%sample%*
CPL	rundll32 shell32.dll, Control_RunDLL "*%sample%*"
SYS	sc create *%random_name%* type= kernel start= demand error= ignore binpath= "*%sample%*" DisplayName= *%random_name%* sc start *%random_name%*
NATIVE APP	rtlrun *%sample%*
MSI	msiexec.exe /i *%sample%*
MHT	winword *%sample%*
XML	msoxmled.exe
RTF, DOC, DOCX, DOCM, DOTM, DOTX, WPS, ODT	winword.exe
XLS, XLSX, XLSM, XLSB, XLAM, XTLX, XTLM, SLK, IQY, ODS	excel.exe
PPT, PPTX, PPTM, PPSX, PPSM, SLDX, SLDM, PPA, PPAM, THMX, POTX, POTM, ODP	powerpnt.exe
ACCDB	msaccess.exe
PUB	mspub.exe
PDF	acrord32.exe
JAR	javaw -jar *%sample%*

Формат файла	Способ запуска
CLASS	java *%sample%*
JS, VBS, WSF, JSE, VBE	wscript /b /nologo *%sample%*
PS1	powershell -file *%sample%*
BAT	cmd /c *%sample%*
SCT	regsvr32.exe /s /i:*%sample%* scrobj.dll
XSL	wmic printjob get /format:"*%sample%*"
MOF	mofcomp *%sample%*
LNK, HTA	*%sample%*
CHM	hh.exe
XLL	excel.exe *%sample%*
ELF	*%sample%*
SH	bash *%sample%*
PY	python *%sample%*
PL	perl *%sample%*
DOCKER	docker load -i *%sample%* docker run *%image_id%*

%sample% — имя файла на виртуальной машине в процессе анализа.

%random_name% — случайным образом сгенерированное имя.

9.2. Как загрузить файл для анализа

Чтобы загрузить файл для анализа

1. На главной странице Dr.Web vxCube нажмите кнопку **Обзор** или поле выбора файла. Выберите файл, который хотите проанализировать.

   Вы также можете перетащить файл в поле выбора файла.

   Формат загружаемого файла определяется автоматически по его содержимому.

Если формат определить не удается (UNK), появится сообщение **Не удалось определить формат файла**. В этом случае вы можете выбрать формат файла вручную.

Форматы MOF, JS, VBS, WSF, JSE, VBE, PS1 и BAT могут быть определены неправильно. Для этих файлов рекомендуем выбирать формат вручную.

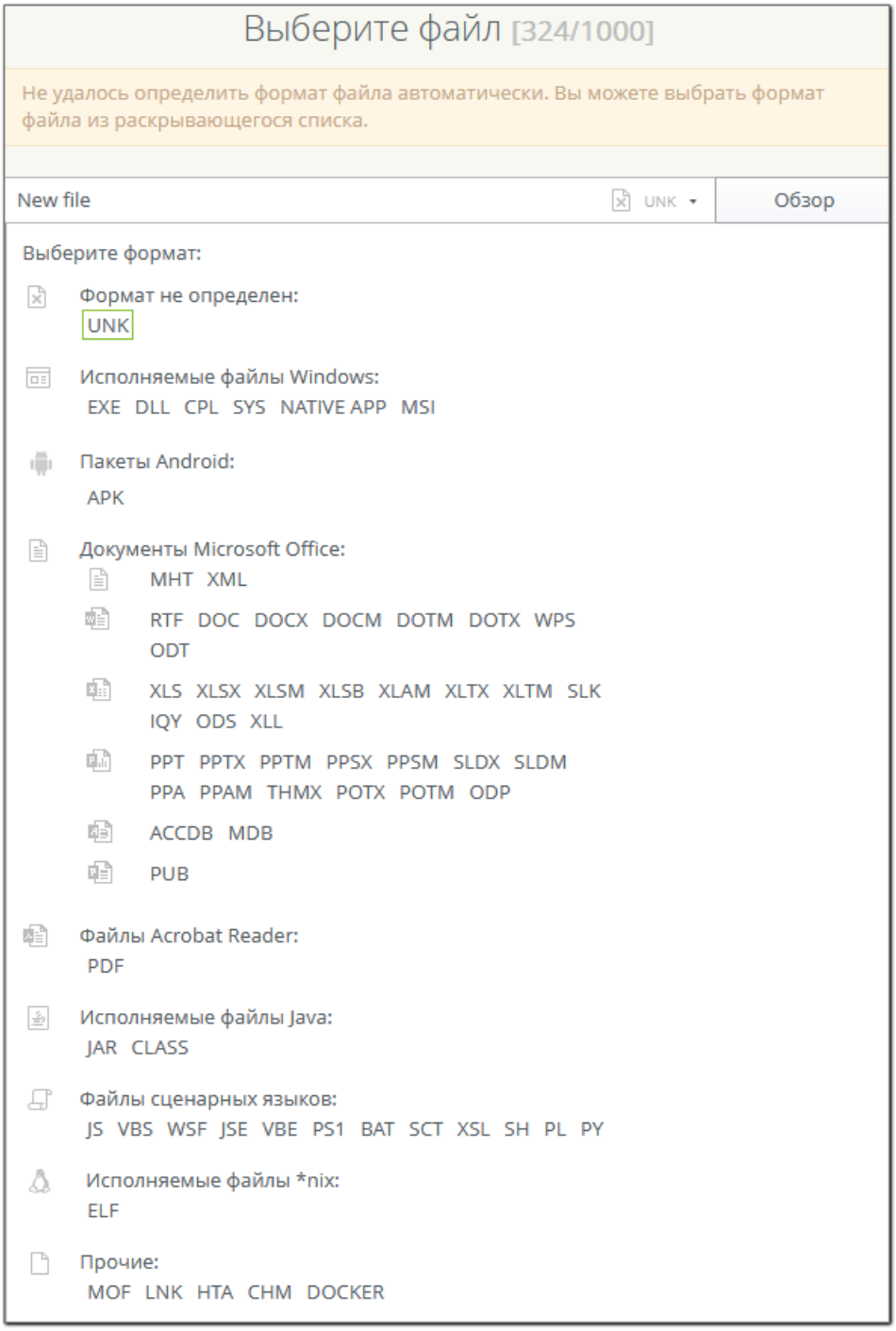


Рисунок 11. Выбор формата файла вручную

Чтобы выбрать формат файла вручную, нажмите стрелку выпадающего списка и выберите нужный формат.

Убедитесь, что выбрали правильный формат. В противном случае результат анализа может быть неточным.

2. Выберите операционную систему или версию приложения для выполнения файла и при необходимости установите дополнительные настройки, чтобы задать особые условия анализа.

   Вы можете выбрать несколько версий ОС или приложений. От этого выбора будет зависеть количество запущенных виртуальных машин. Например, если вы выберете две версии ОС Windows для проверки исполняемого файла (.exe), Dr.Web vxCube запустит две виртуальные машины.

3. Нажмите **Анализировать**, чтобы начать анализ файла.

Вы можете последовательно отправить на анализ несколько файлов. Чтобы выбрать следующий файл, сначала нажмите **Назад** в верхней части страницы, а затем повторите действия. Ход каждой из проверок будет отображаться значком .

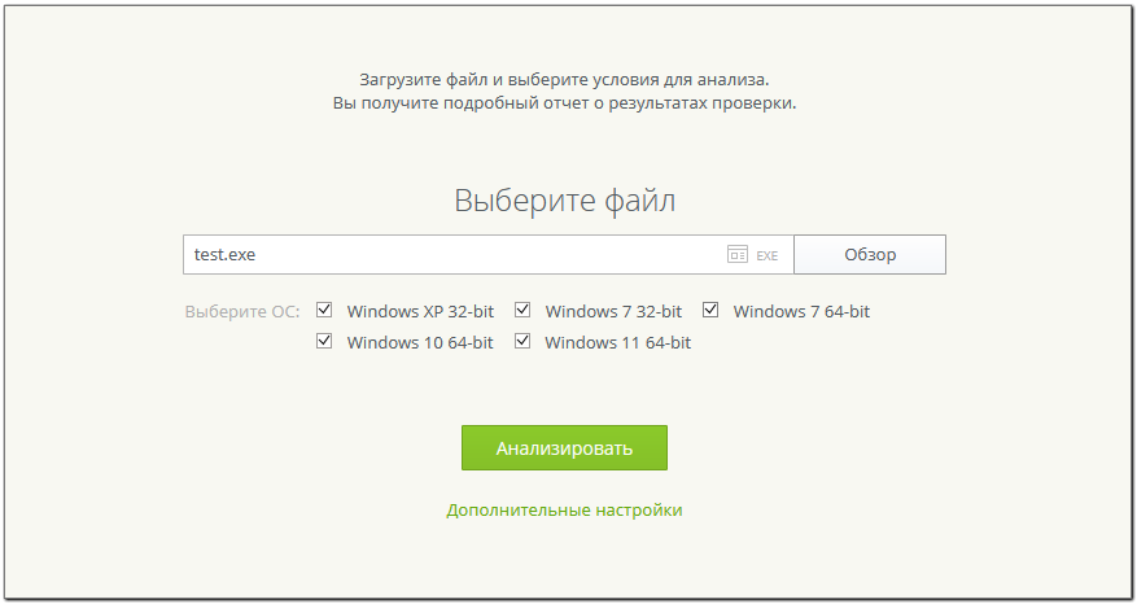


Рисунок 12. Загрузка файла для анализа

9.3. Дополнительные настройки

- **Имя файла**

  Используйте этот параметр, если нужно отправить файл в сервис на анализ под другим именем. При этом исходный файл с прежним именем перезаписан не будет.

- **Использовать VNC**

  Использование VNC-клиента удобно, если вы выбрали более одной операционной системы и хотите влиять на процесс анализа в каждой из них.

  Для активации функции установите флажок **Использовать VNC**. После запуска анализа автоматически открываются дополнительные вкладки браузера. Вкладки подключаются к соответствующим виртуальным машинам через VNC-клиент. На

каждой вкладке в верхней части расположен индикатор выполнения. Индикатор показывает процент завершения и текущее состояние анализа.

Несмотря на то что вкладки в браузере создаются сразу, может потребоваться некоторое время, чтобы подключиться к виртуальным машинам.

> ⚠ Если вы не выбрали этот пункт в окне **Дополнительные настройки** и начали анализ, нажмите **Использовать VNC** на странице анализа. VNC-клиент откроется в новой вкладке.

- **Отслеживать все процессы при использовании VNC**

  По умолчанию этот параметр отключен и в отчет вносятся только те процессы, которые способствовали подозрительной активности.

- **Показывать MITM-трафик**

  Установите этот флажок, чтобы разбирать зашифрованный трафик (доступно только для платформ Windows). По завершении анализа вы сможете просмотреть разобранный трафик. Для этого:

  1. Откройте страницу отчета, созданного по результатам анализа.
  2. Нажмите **Скачать архив**.
  3. Распакуйте архив. Если потребуется, введите пароль, заданный в поле **Пароль для архива отчета** в окне Настройки. Пароль по умолчанию: `vxcube`.
  4. Найдите в распакованном архиве файл `network.pcapng` и загрузите его в программу для анализа сетевого трафика (например, Wireshark).

- **Время выполнения файла**

  По умолчанию время выполнения файла в Dr.Web vxCube — 1 минута. Если необходимо, вы можете сократить или увеличить это время для конкретного файла. Например, если файлу требуется больше времени, чтобы проявить подозрительное поведение, нужно увеличить значение этого параметра, передвинув ползунок вправо.

- **Ограничение на общий размер созданных файлов**

  По умолчанию общий размер всех файлов, созданных во время анализа, не может превышать 64 МБ. Это предельное значение можно увеличить до 512 МБ.

- **Задать команду для запуска файла**

  Эта функция позволяет задать команду запуска анализируемого файла. В качестве команды можно указывать любое приложение из стандартного пакета поставки Windows, например, `rundll32.exe`, `regsvr32.exe`, `notepad.exe` и др. Для использования команды необходимо в поле **Задать команду для запуска файла** указать требуемую команду.

  С помощью специального параметра `%SAMPLE%` можно задать полный путь к анализируемому файлу.

  Этот пункт можно использовать для запуска исполняемых файлов с вызовом специальной экспортируемой функции. Например: `rundll32 %SAMPLE%, ExportedFunction`.

- **Тип подключения**

  По умолчанию используется VPN. Для некоторых типов подключения можно задать адрес прокси-сервера и параметры авторизации. Прокси используется только для TCP-подключений. Трафик других протоколов передается через VPN-сервер по умолчанию. Чтобы перенаправить UDP-трафик, установите флажок **Перенаправлять UDP**.

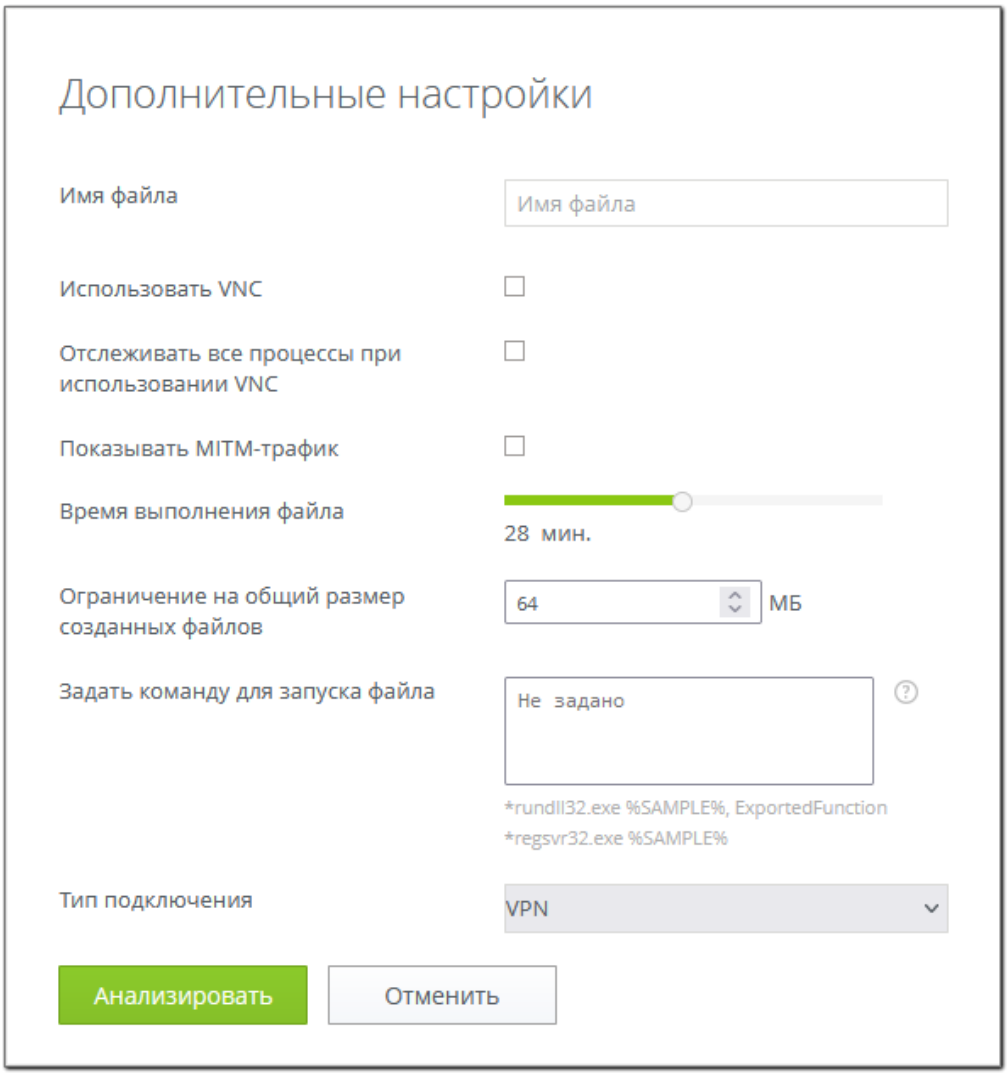


Рисунок 13. Дополнительные настройки

После установки необходимых дополнительных настроек

- Нажмите **Анализировать**, чтобы запустить анализ.
- Нажмите **Отменить**, чтобы сбросить настройки и закрыть окно.

Заданные дополнительные настройки применяются только к текущему файлу. Если вы закрыли окно **Дополнительные настройки** или выбрали другой файл для анализа, задайте дополнительные настройки снова.

10. Отчеты

Полученные в результате анализа данные о выполнении файла группируются в отчет. Вы можете открыть и просмотреть отчет, а также скачать его.

10.1. Как открыть отчет

Чтобы открыть отчет

- Если вы не покидали страницу анализа, отчет откроется автоматически после завершения анализа.
- Если вы покинули страницу до завершения анализа, на главной странице Dr.Web vxCube в разделе **Журнал** выберите необходимый файл.

10.2. Как скачать отчет

На странице отчета доступны кнопки для скачивания, которые позволяют:

Скачать исходный файл.

Скачать архив отчета в формате ZIP. Пароль архива: **vxcube**.

Скачать отчет в формате HTML и PDF.

Скачать файл PCAP.

Чтобы скачать отчет

1. В верхней части страницы отчета выберите платформу.
2. Нажмите **Скачать отчет**. Появится окно **Параметры отчета**.
3. Выберите формат отчета: HTML или PDF.
4. Выберите разделы, которые необходимо включить в отчет. Разделы **Журнал API** и **Намерения** могут содержать тысячи записей, вы можете отфильтровать записи по степени опасности.
5. Нажмите **Скачать отчет**.

Раздел **Намерения** присутствует только в отчетах об анализе пакетов Android.

10.3. Срок хранения отчета

Гарантированный срок хранения отчетов задается в переменной `vxcube_web_keep_reports` (значение по умолчанию: 20 минут). По истечении этого срока отчет может быть удален с сервера.

После удаления отчета на его странице останется только блок с общими сведениями, а вверху появится уведомление о том, что срок хранения отчета истек.

Чтобы заново сформировать отчет, нужно повторно запустить анализ. Для этого на странице отчета нажмите кнопку **Анализировать**.

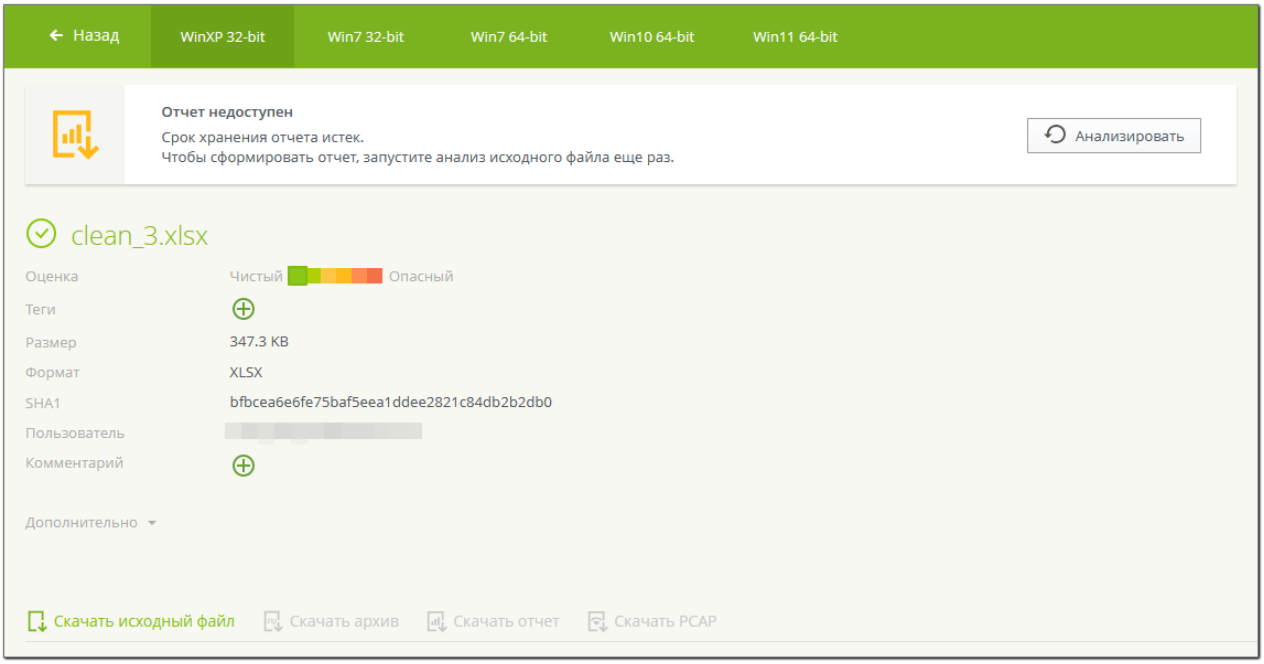


Рисунок 14. Уведомление об истечении срока хранения

10.4. Структура отчета

Отчет разделен на два блока: общие сведения и основную часть.

Общие сведения состоят из двух разделов: *базовые* и *дополнительные сведения*. В разделе базовых сведений указывается размер и формат файла, а также оценка вредоносности файла. В разделе дополнительных сведений приводится, например, такая информация, как имя файла, время начала анализа и его продолжительность. Кроме того, здесь можно увидеть, какие дополнительные параметры были заданы для данного анализа. Изучив эти параметры, вы при необходимости сможете изменить их и проанализировать файл повторно.

В основную часть включены следующие разделы: *Манифест*, *Поведение и правила YARA*, *Граф процессов*, *Описание*, *Файлы и дампы памяти*, *Телефонные звонки и SMS*, *Журнал API и*

намерения, а также *Карта сетевой активности*. В зависимости от типа анализируемого файла список разделов в отчетах может немного отличаться. Например, некоторые разделы будут присутствовать только в отчетах об анализе пакетов Android.

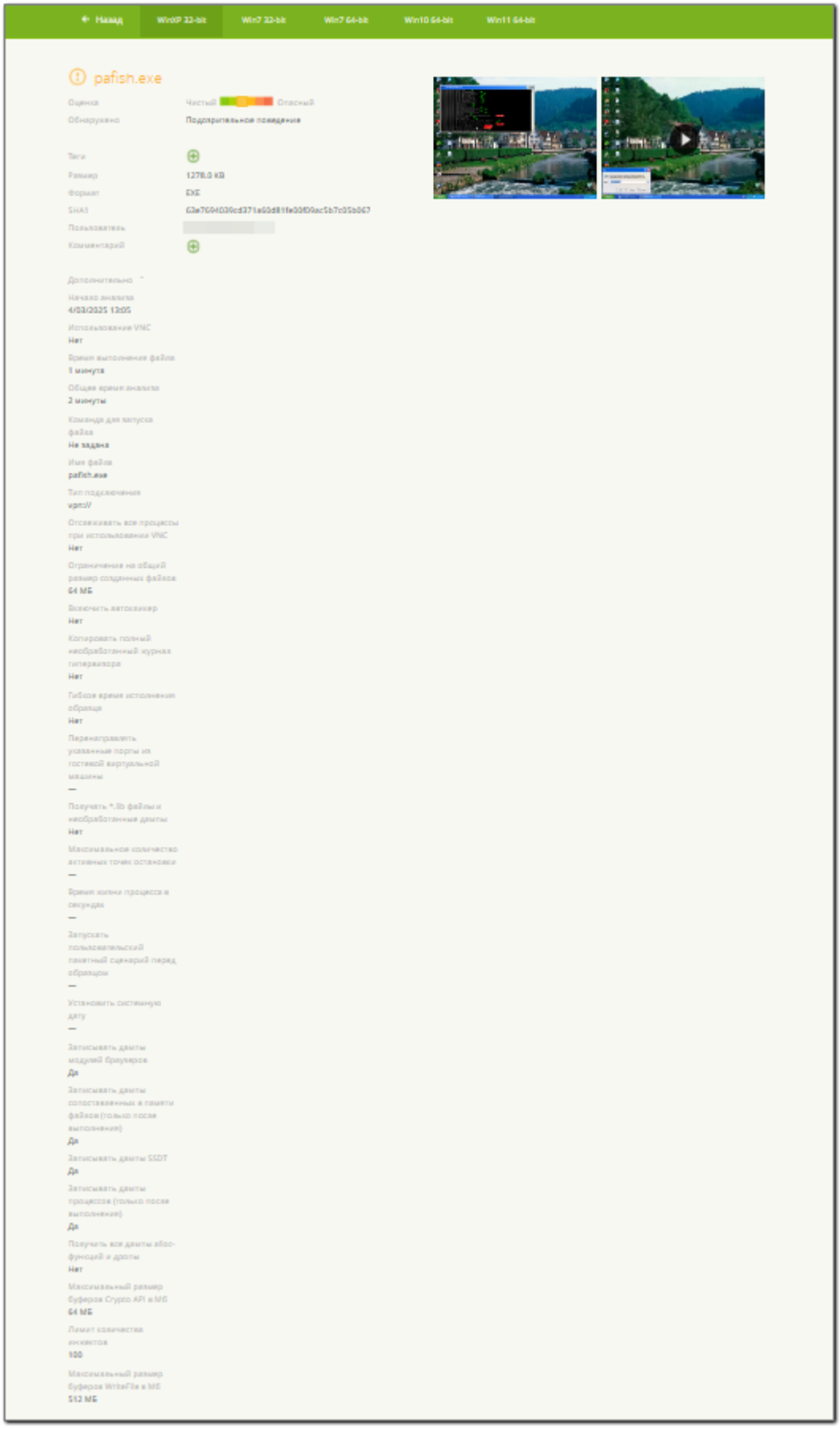


Рисунок 15. Структура отчета

10.4.1. Общие сведения

Пункт	Описание
Оценка	Общая оценка вредоносности файла:  Чистый файл  Подозрительный файл  Опасный файл
Обнаружено	Краткая информация о поведении файла и обнаруженных угрозах.
Теги	Теги, добавленные пользователем или при срабатывании правил YARA.
Размер	Размер файла.
Формат	Формат файла.
SHA1	Хеш файла.
Комментарий	В это поле можно добавить любую нужную вам информацию. Максимальная длина комментария: 200 символов.
Дополнительно	
Начало анализа	Дата и время начала анализа. Началом анализа считается момент, когда файл запустился на виртуальной машине.
Использование VNC	Использовать VNC-клиент во время анализа (да/нет).
Время выполнения файла	Время выполнения файла, заданное в дополнительных настройках анализа.
Общее время анализа	Общая продолжительность анализа файла.
Команда для запуска файла	Команда, заданная в дополнительных настройках, для запуска анализируемого файла.
Имя файла	Имя анализируемого файла. Подробнее...
Тип подключения	Тип подключения. Подробнее...

Пункт	Описание
Отслеживать все процессы при использовании VNC	Отслеживать все процессы при использовании VNC (да/нет). Подробнее...
Ограничение на общий размер созданных файлов	Максимально допустимый общий размер созданных файлов. Подробнее...
Включить автокликер	Включить автокликер (да/нет).
Копировать полный необработанный журнал гипервизора	Копировать полный журнал гипервизора (да/нет).
Гибкое время исполнения образца	Использовать гибкое время исполнения образца (да/нет).
Перенаправлять указанные порты из гостевой виртуальной машины	Перенаправление портов из гостевой виртуальной машины. Пример: `2343, 4353:tcp`.
Получать `*.lib` файлы и необработанные дампы	Получение `*.lib` файлов и необработанных дампов (да/нет).
Максимальное количество активных точек остановки	Указание максимального количества активных точек остановки.
Время жизни процесса в секундах	Время жизни процесса. Пример: `notepad.exe,35,winword.exe,20`.
Запускать пользовательский пакетный сценарий перед образцом	Запуск пользовательского пакетного сценария перед запуском образца.
Установить системную дату	Установка системной даты на виртуальной машине, на которой проводится анализ. Пример: `17.03.2022`.
Записывать дампы модулей браузеров	Записывать дампы модулей браузеров (да/нет).

Пункт	Описание
Записывать дампы сопоставленных в памяти файлов (только после выполнения)	Записывать дампы сопоставленных в памяти файлов (да/нет).
Записывать дампы SSDT	Записывать дампы SSDT (да/нет).
Записывать дампы процессов (только после выполнения)	Записывать дампы процессов (да/нет).
Получить все дампы alloc-функций и дропы	Получить все дампы alloc-функций и дропы (да/нет).
Максимальный размер буферов Crypto API в Мб	Установить максимальный размер буферов Crypto API. Пример: `512`.
Лимит количества инжектов	Установить лимит количества инжектов. Пример: `100`.
Максимальный размер буферов WriteFile в Мб	Установить максимальный размер буферов WriteFile. Пример: `256`.

Справа от общих сведений расположен снимок экрана и видеоотчет о поведении файла на гостевой операционной системе.

10.4.2. Основная часть

Основная часть отчета содержит следующие разделы, наличие которых зависит от формата анализируемого файла.

Раздел	Пакеты Android (опционально)	Другие форматы
Манифест	+	–
Поведение и правила YARA	+	+
Граф процессов	–	+
Описание	+	+
Файлы и дампы памяти	+	+

Раздел	Пакеты Android (опционально)	Другие форматы
Телефонные звонки и SMS	+	–
Журнал API и намерения	+	Только **Журнал API**
Карта сетевой активности	+	+

10.4.2.1. Манифест (опционально)

Раздел присутствует только в отчетах об анализе пакетов Android.

Раздел содержит следующую информацию из файла `AndroidManifest.xml`:

Компонент	Комментарий
Пакет	Имя пакета приложения.
Имя приложения	Имя приложения, которое видит пользователь.
Код версии	Внутренний номер версии.
Имя версии	Название и/или номер версии приложения, которые видит пользователь.
Разрешения	Разрешения, которые приложение запрашивает для своей работы.

В разделе также перечислены следующие компоненты приложения, которые объявляются в манифесте: операции, приемники широковещательных сообщений и службы.

10.4.2.2. Поведение и правила YARA

Раздел содержит две таблицы: **Поведение** и **Правила YARA**. Чтобы открыть интересующую таблицу, нажмите ее название.

Поведение

Раздел содержит краткую информацию об активности файла.

Dr.Web vxCube отслеживает действия, зарегистрированные в процессе анализа файла на виртуальной машине, и распределяет их по категориям в зависимости от степени их вредоносности.

Dr.Web vxCube определяет 3 категории поведения файла:

- Вредоносное.
- Подозрительное.
- Нейтральное.

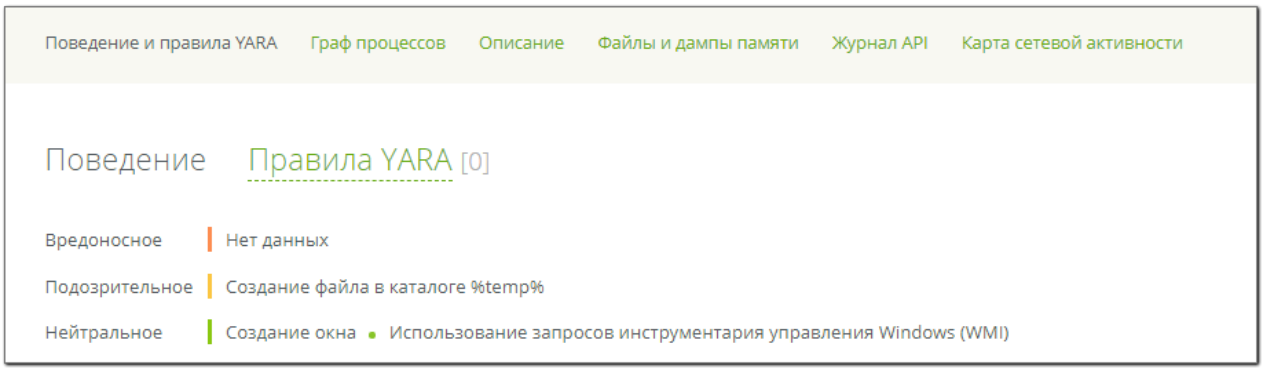


Рисунок 16. Отчет о поведении файла и срабатываниях правил YARA

Правила YARA

Раздел содержит информацию о срабатывании правил YARA. Справа от названия таблицы указано количество правил, сработавших в ходе анализа.

В таблице отображаются результаты проверки, теги и имена сработавших правил.

Чтобы открыть интересующее правило, нажмите его имя.

Чтобы отсортировать столбцы таблицы по возрастанию или убыванию, нажмите заголовки столбцов.

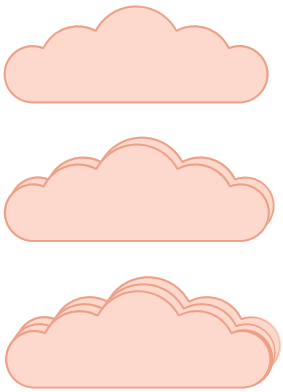
10.4.2.3. Граф процессов

Этого раздела нет в отчетах об анализе пакетов Android.

Раздел содержит информацию о том, какие процессы проявили вредоносную активность во время запуска файла на гостевой операционной системе. Информация представлена в виде интерактивного графа с поясняющим блоком для каждого процесса.

Чтобы открыть граф в новой вкладке браузера, нажмите заголовок **Граф процессов**. Чтобы увеличить или уменьшить масштаб, нажмите + или −. Вы также можете увеличить масштаб, дважды нажав граф.

Условные обозначения

Обозначение	Комментарий
	Степень вредоносности процесса или ресурса. Определяется по шкале от 0 до 100: Менее 20. Менее 40. Менее 60. Менее 80. Менее 100.
	Процесс. Цвет блока соответствует степени вредоносности процесса.
	Сетевой ресурс, к которому осуществляется удаленный доступ. Цвет облака соответствует степени вредоносности ресурса. Внутри облака указывается уровень протокола и IP-адрес удаленного ресурса. Отображается 2 облака, если процесс подключается к ресурсу 2–5 раз. Отображается 3 облака, если процесс подключается 6 и более раз. В этих случаях внутри облака также указывается количество подключений.
	Исходный файл. Значком помечается первый запущенный процесс.
	Известная угроза, содержащаяся в вирусных базах Dr.Web. Значком помечается процесс, в дампе которого обнаружена угроза.
	Известная угроза, содержащаяся в вирусных базах Dr.Web, обнаруженная в дампе подгружаемого модуля. Значком помечается процесс, в который подгружается вредоносный модуль. Если угрозы обнаружены и в дампах процесса, и в дампах модуля, процесс помечается только значком  .
	Создание процесса.
	Инжект в другой процесс.
	Запрос в интернет.
	Запрос RPC.

Поясняющий блок

Нажмите блок процесса, чтобы вывести информацию о нем в поясняющий блок.

Параметры процессов

Параметр	Описание
PID	Уникальный идентификатор процесса.
Полный путь	Путь, по которому запускается процесс.
Параметры запуска	Особенные параметры запуска процесса. Необязательное поле.
Поведение	Правила, соответствующие меткам о подозрительном поведении процесса.
Посмотреть активность процесса	Ссылка на журнал API, данные в котором отфильтрованы по данному процессу. Подробнее об этом пункте можно узнать в разделе Журнал API.
Скачать дамп	Ссылка на загрузку дампа процесса.

Параметры сетевых ресурсов

Параметр	Описание
Адрес	IP-адрес сетевого ресурса.
Порт	Номер порта.

Уровень протокола	Уровень протокола сетевой модели OSI, использованный для передачи данных:
	• **Транспортный**.
	• **Прикладной**.
	Если анализатор не смог распознать протокол прикладного уровня, в этом пункте появится следующая информация:
	`Прикладной: UNK`
	`Нераспознанные данные: {16,03,01,00,41,45…06,00,13,00,00,63,01,00}`

Запрос	DNS-запрос. Этот пункт отображается, если **Уровень протокола** определен как **Прикладной: DNS**.
URL	URL-запрос. Этот пункт отображается, если **Уровень протокола** определен как **Прикладной: HTTP**.

10.4.2.4. Описание

Раздел содержит подробную информацию о подозрительной активности файла, включая список задействованных объектов, подключений и т. д. Информация сгруппирована по категориям и подкатегориям, исходя из поведения конкретного файла. Ниже приведен список категорий и подкатегорий.

Обеспечение автозапуска и распространения

- Модифицирует перечисленные ключи реестра.
- Создает или изменяет перечисленные файлы.
- Устанавливает автозапуск для службы.
- Создает перечисленные сервисы.
- Изменяет перечисленные исполняемые системные файлы.
- Подменяет перечисленные исполняемые системные файлы.
- Заменяет системные бинарные файлы.
- Заменяет системные бинарные файлы с помощью символической ссылки.
- Заражает перечисленные исполняемые файлы.
- Создает перечисленные файлы на съемном носителе.
- Модифицирует главную загрузочную запись (MBR).
- Создает или изменяет файлы для автозапуска:
  - в /init.d;
  - в /router;
  - в /cron;
  - на рабочем столе;
  - в других каталогах.
- Создает или изменяет файлы для автозапуска с помощью символических ссылок:
  - в /cron.
- Создает или изменяет символические ссылки для автозапуска:
  - в /init.d;
  - на рабочем столе;
  - в других каталогах.

Вредоносные функции

- Для обхода брандмауэра удаляет или модифицирует перечисленные ключи реестра.
- Для затруднения выявления своего присутствия в системе:
  - блокирует отображение:
    - скрытых файлов;

        - расширений файлов.
    - блокирует запуск перечисленных системных утилит:
        - интерпретатора командной строки (CMD);
        - диспетчера задач (Taskmgr);
        - редактора реестра (RegEdit);
        - межсетевого экрана (Брандмауэр Windows);
        - обновлений системы (Windows Update);
        - центра обеспечения безопасности (Security Center);
        - системного антивируса (Защитник Windows).
    - блокирует:
        - компонент восстановления системы (SR);
        - систему защиты файлов операционной системы Windows (WFP);
        - средство контроля пользовательских учетных записей (UAC);
        - средство проверки системных файлов (SFC);
        - центр обеспечения безопасности (Security Center);
        - центр поддержки Windows (Action Center).
    - изменяет перечисленные системные настройки:
        - изменяет DNS-сервер;
        - отключает уведомления панели задач.
    - удаляет теневые копии разделов;
    - добавляет исключения антивируса с помощью перечисленных ключей реестра.
- Создает и запускает на исполнение перечисленные процессы:
    - создает и запускает на исполнение (эксплойт);
    - создает и загружает библиотеки (эксплойт);
    - загружает файлы и запускает на исполнение.
- Запускает на исполнение перечисленные процессы.
- Внедряет код в перечисленные процессы:
    - перечисленные системные процессы;
    - перечисленные пользовательские процессы;
    - большое количество пользовательских процессов.
- Устанавливает процедуры перехвата следующих сообщений:
    - о нажатии клавиш клавиатуры:
        - библиотека-обработчик для всех процессов;
        - библиотека-обработчик для процесса.
- Завершает или пытается завершить:

  - процессы;
  - перечисленные системные процессы;
  - перечисленные пользовательские процессы;
  - большое количество пользовательских процессов;
  - процессы приложений анализа трафика или выполнения программ;
  - процессы с определенным именем.
- Ищет ветки реестра, отвечающие за хранение паролей сторонними программами.
- Выполняет операции WMI.
- Регистрирует фильтр файловой системы.
- Ищет перечисленные окна с целью:
  - обхода различных антивирусов;
  - обхода системы защиты файлов Windows (WFP);
  - обнаружения утилит для анализа;
  - обнаружения различных программ и игр;
  - обнаружения виртуальных машин.
- Создает onion-сервис.
- Загружает перечисленные драйверы.
- Перехватывает перечисленные функции в SSDT (System Service Descriptor Table):
  - драйвер-обработчик.
- Устраняет перехваты функций в SSDT (System Service Descriptor Table).
- Перебирает пароли аккаунтов ОС.
- Проводит атаку перебором по сети.
- Отключает AMSI.
- Изменяет настройки брандмауэра.
- Изменяет настройки маршрутизатора.
- Останавливает системные службы.
- Управляет службами.
- Блокирует через брандмауэр:
  - SSH;
  - telnet;
  - стандартные веб-порты.
- Изменяет перечисленные настройки проводника Windows (Windows Explorer).
- Изменяет перечисленные настройки браузера Windows Internet Explorer.
- Влияет на процессы:
  - скрывает перечисленные процессы;

  - выполняет трассировку процессов;
  - встраивается в процессы.
- Принудительно разрешает автозапуск со съемных носителей.
- Без разрешения пользователя устанавливает новую стартовую страницу для Internet Explorer.
- Пытается завершить работу операционной системы Windows.
- Отправляет SMS.
- Выполняет код детектируемых угроз.
- Загружает из интернета детектируемые угрозы.
- Отправляет данные о контактах устройства на удаленный сервер.
- Отправляет данные входящих SMS на удаленный сервер.
- Перекрывает экран собственным окном, блокируя доступ к интерфейсу.
- Устанавливает пароль на экран блокировки.
- Предлагает установить стороннее приложение.
- Скрывает свой значок с экрана.
- Завершает входящие телефонные звонки.
- Приглушает входящие телефонные звонки.
- Перехватывает входящие SMS и не позволяет передавать их обработчикам других приложений.
- Деактивирует администратора устройства.
- Удаляет данные пользователя.
- Угроза, выявленная на основе машинного обучения.
- Содержит типичный для банковских троянов и вирусов код.
- Содержит типичный для локеров код.
- Загружает для исполнения перечисленные выявляемые угрозы.
- Загружает из интернета перечисленные выявляемые угрозы.
- Запускает большое число процессов.

Изменения в файловой системе

- Создает перечисленные файлы.
- Присваивает атрибут «скрытый» перечисленным файлам.
- Удаляет перечисленные файлы.
- Помечает записанный файл как исполняемый.
- Помечает файл как исполняемый.
- Удаляет файл.
- Удаляет системный бинарный файл.
- Создает или изменяет символические ссылки.

- Записывает в системную область:
  - файлы;
  - символические ссылки.
- Записывает в поддиректорию системной области:
  - файлы;
  - символические ссылки.
- Записывает во временную поддиректорию:
  - файлы;
  - символические ссылки.
- Создает директории:
  - в поддиректории системной директории;
  - в системной директории;
  - в поддиректории временной директории;
  - во временной директории;
  - в других директориях.
- Удаляет директории:
  - из системной директории;
  - из поддиректории системной директории;
  - из поддиректории временной директории;
  - из других директорий.
- Перемещает перечисленные системные файлы.
- Перемещает перечисленные файлы.
- Подменяет перечисленные исполняемые файлы.
- Изменяет файл HOSTS.
- Подменяет файл HOSTS.
- Самоперемещается.
- Самоудаляется.
- Создает файлы.
- Изменяет права доступа:
  - файла;
  - записанного файла.
- Изменяет владельца:
  - файла;
  - записанного файла.
- Устанавливает блокировку на файлы.

- Изменяет время создания, доступа или изменения файлов.
- Монтирует файловые системы.
- Демонтирует файловые системы.
- Создает файлы с требованием оплатить расшифровку файлов (Trojan.Encoder).
- Изменяет множество файлов пользовательских данных (Trojan.Encoder).
- Изменяет расширения файлов пользовательских данных (Trojan.Encoder).
- Задает разрешения на выполнение файлов.
- Добавляет исключения в Microsoft Defender.

Сетевая активность

- Подключается к сетевому ресурсу.
- Открывает порт.
- Отправляет данные на сервер.
- Получает данные от сервера.
- Получает доступ к SSH.
- Связывается с сервером по протоколу:
  - HTTP;
  - IRC.
- TCP:
  - запросы HTTP GET;
  - запросы HTTP POST;
  - запросы HTTP HEAD;
  - запросы HTTP PATCH;
  - запросы HTTP PUT;
  - запросы HTTP DELETE;
  - запросы HTTP OPTIONS;
  - запросы HTTP TRACE;
  - запросы HTTP неизвестного формата.
- UDP:
  - запросы DNS.

Другое

- Добавляет корневой сертификат.
- Отключает сертификат.
- Собирает информацию:
  - об ОС;
  - о ЦП;

  - об оперативной памяти;
  - о сетевой активности.
- Изменяет значение AutoConfigURL на указанное.
- Подменяет имя приложения.
- Ищет перечисленные окна.
- Создает и исполняет файлы.
- Файл защищен упаковщиком Themida компании Oreans Technologies.
- Использует альтернативные потоки данных NTFS.
- Загружает драйверы.
- Выгружает модуль ядра.
- Устанавливает модуль ядра на автозагрузку.
- Запускает shell-скрипты.
- Запускается как фоновая программа (демон).
- Компилирует исходный код.
- Читает информацию из /proc/kallsyms.
- Загружает динамические библиотеки.
- Совершает телефонные звонки.
- Использует алгоритмы для шифрования данных.
- Использует алгоритмы для расшифровки данных.
- Использует повышенные привилегии.
- Использует права администратора.
- Получает права суперпользователя.
- Осуществляет доступ к приватному интерфейсу ITelephony.
- Использует специальную библиотеку для скрытия исполняемого байт-кода.
- Содержит функциональность для автоматической отправки SMS.
- Осуществляет доступ к интерфейсам записи аудио/видео.
- Записывает аудио/видео.
- Осуществляет доступ к интерфейсу камеры.
- Изменяет настройки громкости и вибрации.
- Получает информацию о местоположении устройства.
- Получает информацию о сети.
- Получает информацию о телефоне (номере, IMEI и т. д.).
- Получает информацию о настройках APN.
- Получает информацию об активных администраторах устройства.
- Получает информацию об установленных приложениях.

- Получает информацию о запущенных приложениях.
- Получает информацию о привязанных к устройству аккаунтах.
- Добавляет задания в системный планировщик.
- Отрисовывает собственные окна поверх других приложений.
- Обрабатывает информацию из SMS-сообщений.
- Получает информацию о входящих/исходящих звонках.
- Получает информацию об отправленных/принятых SMS.
- Получает информацию о телефонных контактах.
- Включает/отключает все камеры.
- Управляет Wi-Fi-подключением.
- Проверяет наличие антивирусных приложений.
- Перехватывает уведомления.
- Запрашивает разрешение на отображение системных уведомлений.
- Образец из Google Play Store.
- Перезапускает анализируемый образец.

10.4.2.5. Файлы и дампы памяти

Раздел содержит две таблицы: **Созданные файлы** и **Дампы памяти**. Справа от названия каждой таблицы указано количество объектов, обнаруженных в ходе анализа.

Чтобы открыть интересующую таблицу, нажмите ее название.

Чтобы отсортировать столбцы таблицы по возрастанию или убыванию, нажмите заголовки столбцов.

Чтобы скачать файл из таблицы, нажмите значок **Скачать файл**  . Если Dr.Web vxCube не собрал файл из-за ограничения ресурсов, файл не получится скачать. В этом случае отображается значок  .

Созданные файлы

Таблица содержит информацию о файлах, созданных в процессе анализа. В таблице отображаются путь, хеш и имя обнаруженной угрозы.

Дампы памяти

Таблица содержит информацию о следующих объектах:

- Дампы памяти.
- Инжекты.

- Блоки памяти, выделеные исходным файлом во время его выполнения. Выделенная память может содержать следы вредоносной активности.

В таблице отображаются имя файла, хеш, уникальный идентификатор процесса (PID) и имя обнаруженной угрозы.

Имя обнаруженной угрозы отображается в таблицах при условии, что она содержится в базе данных Dr.Web.

10.4.2.6. Телефонные звонки и SMS (опционально)

Раздел присутствует только в отчетах об анализе пакетов Android.

Раздел содержит информацию об исходящих телефонных звонках и SMS-сообщениях, которые были совершены анализируемым приложением. В таблице указаны телефонные номера получателей и тексты сообщений.

10.4.2.7. Журнал API и намерения

Раздел содержит две таблицы: **Журнал API** и **Намерения**.

Таблица **Намерения** присутствует только в отчетах об анализе пакетов Android.

Справа от названия каждой таблицы указано количество объектов, обнаруженных в ходе анализа.

Чтобы открыть интересующую таблицу, нажмите ее название.

Чтобы отсортировать столбцы таблицы по возрастанию или убыванию, нажмите заголовки столбцов.

Чтобы отфильтровать данные в таблицах по вредоносности, нажмите один из цветов в шкале  . Фильтр работает по принципу включения более высокого уровня вредоносности в предыдущий уровень.

Журнал API

В таблице **Журнал API** собрана информация обо всех событиях, произошедших на виртуальной машине во время запуска файла. **Журнал API** представляет собой структурированную в таблицу информацию из раздела Граф процессов.

Нажмите **Открыть журнал API в новой вкладке**, чтобы открыть раздел в новой вкладке браузера.

Параметр	Комментарий
Время	Время события. Отсчитывается с момента запуска анализа файла.
Процесс	Полный путь к процессу на гостевой операционной системе.
Событие	Событие, произошедшее во время запуска файла. Соответствует общеупотребляемым API-функциям.
Аргументы	Аргументы событий. Указывают на особые условия выполнения события.

Намерения

В таблице **Намерения** перечислены намерения, которые были отправлены анализируемым приложением, чтобы запустить компоненты других приложений.

Параметр	Комментарий
Время	Время действия. Отсчитывается с момента запуска анализа файла.
Данные	Данные, с которыми выполняется действие.
Действие	Название выполняемого действия.
Транзакция	Транзакция, определяющая тип запускаемого компонента: • START_ACTIVITY — запуск операции. • START_SERVICE — запуск службы. • BROADCAST_INTENT — рассылка широковещательных сообщений.
Имя компонента	Компонент, который получает намерение.

10.4.2.8. Сетевая активность

В разделе **Сетевая активность** содержится информация о подключениях, которые были инициированы во время выполнения файла. Эта информация представлена в виде карты и таблицы. На карте указаны сведения о количестве подключений и конечных точках. В таблице под картой вы найдете следующие данные по каждому подключению:

- **Время:** время с захвата первого пакета (в секундах).
- **Протокол:** протокол, используемый для подключения
- **Отправитель:** IP-адрес отправителя пакета.
- **Получатель:** IP-адрес получателя пакета.
- **Информация:** информация о передаваемом пакете.

Вы можете сортировать информацию по возрастанию и убыванию во всех столбцах, кроме столбца **Информация**. Для этого нажмите заголовок столбца. Справа от заголовка столбца, по которому на данный момент сортируются подключения, отображается значок ▲ или ▼. Чтобы изменить направление сортировки, нажмите заголовок еще раз.

По умолчанию в разделе **Сетевая активность** отображаются только те подключения, которые инициировал анализируемый файл. Чтобы дополнительно показывать подключения, которые инициировали вы через VNC-клиент, перед запуском анализа установите в окне Дополнительные настройки флажок **Отслеживать все процессы при использовании VNC**.

10.5. Журнал анализа файлов

Журнал содержит информацию об анализах, проведенных ранее, и расположен на главной странице Dr.Web vxCube под блоком выбора файла.

Журнал позволяет:

- Осуществлять поиск по строке, фильтровать и сортировать записи.
- Проверять прогресс текущего анализа.
- Просматривать, удалять и скачивать отчеты проанализированных файлов.

Управление журналом

Чтобы задать количество записей, отображаемых на одной странице

- Нажмите выпадающее меню под таблицей.

Чтобы отсортировать записи

- Нажмите заголовок соответствующего столбца.

  Вы можете отсортировать записи по логину пользователя, имени файла или дате.

Чтобы отфильтровать записи

- Введите строку в поле поиска. Поиск осуществляется по всем столбцам таблицы.
- Нажмите **Журнал** для фильтрации по типу файла.

Чтобы выбрать отображаемые столбцы

- В правом углу таблицы нажмите ⋯.
- Выберите нужные для отображения столбцы.

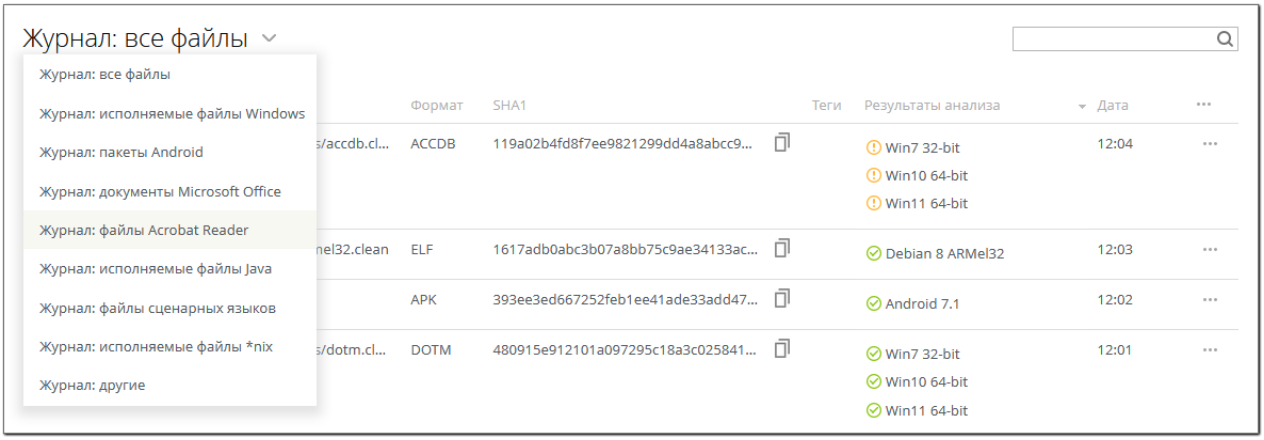


Рисунок 17. Выбор типа файлов в Журнале

Чтобы открыть страницу отчета об анализе

- Нажмите имя интересующего вас файла.

Чтобы скачать отчет об анализе

- Для соответствующего файла наведите курсор на значок ⋯ и выберите **Скачать архив**. Подробный отчет будет скачан в формате ZIP.

Чтобы удалить отчет об анализе

- Для соответствующего файла наведите курсор на значок ⋯ и выберите **Удалить отчет**.

Рисунок 18. Действия, доступные для проверенных файлов в Журнале

10.6. Теги

Чтобы работать с отчетами было удобнее, вы можете использовать в них специальные метки классификации, *теги*. Добавить теги можно двумя способами:

- При добавлении правила YARA. Тогда, если это правило сработает в процессе анализа, указанные теги автоматически попадут в отчет.
- Вручную в готовый отчет. Для этого:
  1. В строчке отчета **Теги** нажмите ⊕.
  2. Введите имя тега, используя латинские буквы, цифры или подчеркивание.

3. Нажмите +.

11. API

API Dr.Web vxCube позволяет:

- Отправлять файлы на анализ без участия человека.
- Отправлять больше файлов за меньшее количество времени.
- Систематизировать результаты программным путем.

Для работы с API Dr.Web vxCube рекомендуем использовать наш API-клиент Dr.Web vxCube API Client. В этом случае вам не придется самостоятельно формировать запросы для таких действий, как, например, отправка образцов на анализ, получение результатов и скачивание отчетов.

В настоящее время используется API Dr.Web vxCube версии 2.0. Эта версия API поддерживает только формат JSON. Для всех запросов к API используйте базовый URL-адрес:

```
https://<IP сервера/доменное имя сервера>/api-2.0/
```

11.1. Аутентификация

Каждый API-запрос к сервису Dr.Web vxCube по API должен быть аутентифицирован с помощью API-ключа. Этот ключ служит идентификатором пользователя, своеобразным ключом доступа к сервису (по аналогии с логином и паролем в веб-интерфейсе). API-ключ указывается в заголовке `Authorization` API-запроса.

Пример запроса

```
curl -X GET https://<IP-адрес/доменное имя сервера>/api-2.0/analyses/60e21c98-7c2a-4112-81b5-a577f6cdf4db \
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee"
```

Вы можете создать API-ключ в веб-интерфейсе сервиса или с помощью API-запроса.

11.2. Управление API-ключами

Вы можете создавать API-ключи, просматривать их и удалять.

У вас может быть не более 10 API-ключей. Если вы хотите добавить новый API-ключ, но у вас уже есть 10 ключей, сначала удалите один или несколько из имеющихся.

Чтобы создать API-ключ в веб-интерфейсе

1. В правом верхнем углу главной страницы Dr.Web vxCube нажмите **Профиль** > **Настройки**.
2. Выберите слева вкладку **API-сессии**.
3. В поле **Новый ключ** введите имя ключа и нажмите +. Ключ появится в списке **Существующие ключи**.
4. Вы можете скопировать созданный ключ, нажав справа от него значок.

Чтобы создать API-ключ с помощью API-запроса

- Отправьте запрос POST login.

Чтобы посмотреть список имеющихся у вас API-ключей

1. В правом верхнем углу главной страницы Dr.Web vxCube нажмите **Профиль** > **Настройки**.
2. Выберите слева вкладку **API-сессии**.
3. В списке **Существующие ключи** вы увидите свои API-ключи.

> Если у вас уже есть API-ключ, вы можете получить его в ответе на API-запрос POST login.

Чтобы удалить API-ключ

1. В правом верхнем углу главной страницы Dr.Web vxCube нажмите **Профиль** > **Настройки**.
2. Выберите слева вкладку **API-сессии**.
3. В списке **Существующие ключи** справа от ключа, который хотите удалить, нажмите .

> Вы можете отменить удаление ключа. Для этого нажмите **Восстановить** рядом с информацией о его удалении. Но если вы закроете окно **Настройки**, то кнопка **Восстановить** исчезнет и ключ будет удален навсегда.

11.3. Эндпоинты

11.3.1. analyses

Эндпоинт используется для управления анализом.

DELETE analyses/<analysis_id:uuid>

Описание	Параметры	Результат
Удалить анализ.	—	Анализ удален, код 204.

GET analyses

Описание	Результат
Получить данные об анализах.	Список объектов Analysis.

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`format_group_name`	string	Фильтрация по типу файла.	Нет

GET analyses/<analysis_id:uuid>

Описание	Параметры	Результат
Получить подробную информацию об анализе.	—	Объект Analysis.

Пример использования

GET analyses/<analysis_id:uuid>/archive

Описание	Параметры	Результат
Скачать полный архив с результатами анализа.	—	Архив, содержащий результаты анализа по всем задачам.

Пример использования

GET analyses/<analysis_id:uuid>/sample

Описание	Параметры	Результат
Скачать файл, отправленный на анализ.	—	Файл, отправленный на анализ.

POST analyses

Описание	Результат
Запустить анализ файла.	Объект Analysis.

Параметры

Параметр	Тип	Описание	Обязательный
`analysis_time`	integer	Время выполнения файла в секундах, от 30 до 300. По умолчанию `analysis_time=60`.	Нет
`convert_video`	boolean	Преобразовывать видео в процессе анализа.	Нет
`copylog`	boolean	Копирование полного необработанного журнала гипервизора.	Нет
`crypto_api_limit`	integer	Максимальный размер буферов Crypto API в Мб.	Нет
`custom_cmd`	string/null	Команда для выполнения файла.	Нет
`drop_size_limit`	integer	Ограничение на общий размер созданных файлов.	Нет
`dump_browsers`	string	Записывать дампы модулей браузеров.	Нет
`dump_mapped`	boolean	Записывать дампы сопоставленных в памяти файлов (только после выполнения).	Нет
`dump_processes`	boolean	Записывать дампы процессов (только после выполнения).	Нет
`dump_size_limit`	integer	Максимальный размер коллекционируемых дропов.	Нет
`dump_ssdt`	boolean	Записывать дампы SSDT.	Нет

Параметр	Тип	Описание	Обязательный
`flex_time`	boolean	Гибкое время исполнения образца.	Нет
`format_name`	string	Формат файла.	Да, если формат не определен автоматически
`forwards`	array [string]/null	Перенаправление указанных портов из гостевой виртуальной машины.	Нет
`generate_cureit`	boolean	Подготовить утилиту Dr.Web CureIt! для обезвреживания угроз в исходном файле и во всех файлах, созданных во время анализа.	Нет
`get_lib`	boolean	Получать `*.lib` файлы и необработанные дампы.	Нет
`injects_limit`	integer	Лимит количества инжектов.	Нет
`monkey_clicker`	boolean	Включить автокликер.	Нет
`net`	string	Команда для перенаправления сетевого трафика виртуальной машины в соответствии с указанными настройками. • `VPN = vpn://` (используется по-умолчанию, если не указан параметр net) • `TOR = tor://` • `Socks4 = socks4://host:port` • `Socks5 = socks5://[login:password@]host:port?parameters` • `Shadowsocks = shadowsocks://[login:password@]host:port?parameters` Возможные значения `parameters`: `udp` — поведение UDP-протокола (`udp=on` — перенаправлять весь UDP-трафик, `udp=off` — не перенаправлять трафик); `login:password` — параметры авторизации на прокси сервере (не обязательно для Socks5, обязательно для Shadowsocks).	Нет
`no_clean`	boolean	Получить все дампы alloc-функций и дропы.	Нет
`optional_count`	integer/null	Максимальное количество активных точек остановки.	Нет

Параметр	Тип	Описание	Обязательный
`platforms`	array [string]/null	Платформы для выполнения файла.	Нет
`proc_lifetime`	string/null	Время жизни процесса в секундах. Пример: `'notepad.exe,35,winword.exe,20`	Нет
`sample_id`	integer	ID исходного файла.	Да
`set_date`	string	Установить системную дату (формат: 17.03.2022).	Нет
`write_file_limit`	integer	Максимальный размер буферов WriteFile в Мб.	Нет

Пример использования

POST analyses/<analysis_id:uuid>/restart

Описание	Параметры	Результат
Перезапустить все удаленные или неудавшиеся задачи указанного анализа.	—	Перезапуск удаленных или неудавшихся задач.

11.3.2. formats

Эндпоинт используется для получения информации о поддерживаемых форматах.

GET formats

Описание	Параметры	Результат
Получить список форматов, которые поддерживает Dr.Web vxCube.	—	Список объектов Format.

11.3.3. login

Эндпоинт используется, чтобы получить один из созданных ранее API-ключей или создать новый. У вас может быть не более 10 API-ключей.

POST login

Описание	Результат

Получить API-ключ.	`{`
	`"new_key": <true или false>`
	`"api_key": "<API-ключ>"`
	`"start_date": "<дата>"`
	`"name": <имя ключа>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`login`	string	Логин пользователя.	Да
`password`	string	Пароль пользователя.	Да
`new_key`	boolean	Определяет, создавать ли новый API-ключ или использовать один из созданных ранее. По умолчанию `new_key=false`. Если у вас нет созданных API-ключей, можете не указывать этот параметр, API-ключ будет создан все равно.	Нет
`name`	string	Имя ключа.	Нет

Пример использования

11.3.4. platforms

Эндпоинт используется для получения информации о поддерживаемых платформах.

GET platforms

Описание	Параметры	Результат
Получить список поддерживаемых платформ.	—	Список объектов Platform.

Поддерживаемые платформы

В следующей таблице приведены поддерживаемые платформы для разных форматов файлов.

Формат файла	Поддерживаемые платформы
ARJ, BAT, BZ2, CAB, CDF, CHM, CPL, DLL, EML, EXE, GZ, HTA, JS, JSE, LNK, MOF, MSI, Native App, PS1, RAR, SCT, SYS, TAR, VBE, VBS, WSF, XSL, XZ, ZIP, 7Z	winxpx86, win7x86, win7x64, win10x64, win11x64
ACCDB, DOC, DOCM, DOCX, DOTM, DOTX, IQY, MDB, MHT, ODP, ODS, ODT, POTM, POTX, PPA, PPAM, PPSM, PPSX, PPT, PPTM, PPTX, PUB, RTF, SLDM, SLDX, SLK, THMX, XLAM, XLL, XLS, XLSB, XLSM, XLSX, XLTM, XLTX, XML, WPS	office_xp, office_7_32, office_7_64, office_10_64, office_11_64
PDF	acrobat_xp_10, acrobat_7_32_11, acrobat_7_64_15, acrobat_10_64_15, acrobat_11_64_15
CLASS, JAR	java_xp, java_7_32, java_7_64, java_10_64, java_11_64
DOCKER, PL, PY, SH	intel64_astra_ce_2.12, intel64_astra_se_1.7.2, intel64_debian_bullseye
ELF	arm64_debian_bullseye, armel32_debian_jessie, armhf32_debian_bullseye, intel32_debian_bullseye, intel64_astra_ce_2.12, intel64_astra_se_1.7.2, intel64_debian_bullseye, mips32_debian_buster, mipsel32_debian_bullseye, mipsel64_debian_bullseye, ppc32_debian_jessie, ppcel64_debian_bullseye
APK	android7.1

11.3.5. samples

Эндпоинт используется для управления анализируемыми файлами.

GET samples

Описание	Результат
Получить список файлов, загруженных ранее.	Список объектов Sample.

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`md5`	string	Фильтрация по MD5.	Нет
`sha1`	string	Фильтрация по SHA1.	Нет
`sha256`	string	Фильтрация по SHA256.	Нет
`format_name`	string	Фильтрация по формату файла.	Нет
`format_group_name`	string	Фильтрация по типу файла.	Нет

GET samples/<sample_id:number>

Описание	Параметры	Результат
Получить данные о файле, загруженном ранее.	—	Объект Sample.

GET samples/<sample_id:number>/analyses

Описание	Параметры	Результат
Получить данные о проведенных для файла анализах.	—	Объект Analysis.

POST samples

Описание	Результат
Загрузить файл на сервер Dr.Web vxCube.	Объект Sample.

Параметры

Параметр	Тип	Описание	Обязательный
`file`	string	Файл, который нужно загрузить на сервер. Указывается полный путь к файлу с предшествующим символом `@`.	Да
`password`	string	Пароль для загружаемого архива. Допустимая длина пароля: от 1 до 25 символов.	Нет

Пример использования

11.3.6. sessions

Эндпоинт используется для управления сессиями.

DELETE sessions/<api_key:string>

Описание	Параметры	Результат
Удалить сессию с указанным API-ключом.	—	Сессия удалена, код 204.

GET sessions

Описание	Параметры	Результат
Получить список всех открытых сессий.	—	Список объектов Session.

11.3.7. tasks

Эндпоинт используется для управления задачами анализа и данными отчета.

GET tasks/<task_id:number>

Описание	Параметры	Результат
Получить данные о задаче.	—	Объект Task.

GET tasks/<task_id:number>/archive

Описание	Параметры	Результат
Скачать архив с результатами анализа.	—	Архив с результатами анализа.

GET tasks/<task_id:number>/sample

Описание	Параметры	Результат
Скачать файл, отправленный на анализ.	—	Файл, отправленный на анализ.

GET tasks/<task_id:number>/report

Описание	Параметры	Результат
Скачать одностраничный HTML-отчет.	—	Одностраничный HTML-отчет.

GET tasks/<task_id:number>/graph

Описание	Параметры	Результат
Скачать граф в формате SVG.	—	Граф в формате SVG.

GET tasks/<task_id:number>/dumps

Описание	Результат

Получить данные из таблицы Дампы памяти.	{
	"total_count": <*число*>,
	"items": <*список объектов* Dump>
	}

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`search`	string	Образец для поиска по строке.	Нет

GET tasks/<task_id:number>/drops

Описание	Результат

Получить данные из таблицы Созданные файлы.	`{`
	`"total_count": <число>,`
	`"items": <список объектов Drop>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`search`	string	Образец для поиска по строке.	Нет

GET tasks/<task_id:number>/networks

Описание	Результат

Получить данные из таблицы Карта сетевой активности.	`{`
	`"total_count": <число>,`
	`"items": <список объектов Connection>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет

GET tasks/<task_id:number>/api_log

Описание	Результат

Получить данные из таблицы Журнал API.	`{`
	`"total_count": <число>,`
	`"items": <список объектов APIEvent>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`search`	string	Образец для поиска по строке.	Нет

GET tasks/<task_id:number>/intents (опционально)

Описание	Результат

Получить данные из таблицы Намерения. Эндпоинт используется только для задач, запущенных на Android.	`{`
	`"total_count": <число>,`
	`"items": <список объектов Intent>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`search`	string	Образец для поиска по строке.	Нет

GET tasks/<task_id:number>/phone_actions (опционально)

Описание	Результат

Получить данные из таблицы Телефонные звонки и SMS. Используется только для задач, запущенных на Android.	`{`
	`"total_count": <`*число*`>,`
	`"items": <`*список объектов* Call *и* Message`>`
	`}`

Параметры

Параметр	Тип	Описание	Обязательный
`count`	integer	Количество возвращаемых объектов, 1...100. Чтобы получить больше объектов, используйте несколько запросов и параметр `offset`. По умолчанию `count=10`.	Нет
`offset`	integer	Смещение, 0...+∞. По умолчанию `offset=0`.	Нет
`search`	string	Образец для поиска по строке.	Нет

GET tasks/<task_id:number>/archive_storage

Описание	Параметры	Результат

Получить список файлов и директорий в архиве или скачать файл или директорию из архива.	`path` (string) — путь, необязательный параметр	Если `path` не указан:
		`{`
		`"folders": <`*список папок в архиве*`>,`
		`"files": <`*список файлов в архиве*`>`

Описание	Параметры	Результат
		} Если `path` указан — файл или архив папки.

[Пример использования]

POST tasks/<task_id:number>/restart

Описание	Параметры	Результат
Перезапустить удаленную или неудавшуюся задачу.	—	Перезапуск удаленной или неудавшейся задачи.

11.3.8. ws/progress

Чтобы подключиться по протоколу WebSocket и получить данные о ходе анализа в режиме реального времени, в запросе укажите объект JSON в виде строки:

```
{"analysis_id": <ID>}
```

В ответ вы получите объект JSON:

```
{'message': '<сообщение>', 'progress': <ход выполнения>, 'task_id': <ID>}
```

11.4. Объекты

11.4.1. Analysis

Объект **Analysis** содержит общие сведения об анализе и список объектов Task.

Структура

Ключ	Тип	Описание
`id`	UUID	UUID задачи.
`sha1`	string	Хеш SHA1.
`sample_id`	integer	ID анализируемого файла.
`size`	integer	Размер файла в байтах.

Ключ	Тип	Описание
format_name	string/null	Формат файла. Совпадает с форматом Sample.format_name, если формат не был указан явно при запуске анализа файла.
start_date	string (datetime.iso8601)	Дата и время запуска анализа.
user_name	string	Логин пользователя.
tasks	array [Task]	Список задач. Соответствует выбранным платформам.

Примеры

Если вы запросили определенный анализ по его ID, в ответ вы получите объект **Analysis**, в котором ключ `tasks` — это список объектов TaskFinished или TaskProcessing:


```
{
  "id": 1629b17b-fd44-46e6-97a2-1310c1f050a4,
  "sample_id": 6248,
  "size": 3242863,
  "sha1": "8c81eb1b6a87e30656d479968eca969bc59bdeb3",
  "start_date": "2018-12-12T11:29:44.645968+00:00",
  "user_name": "name_example",
  "format_name": "rtf",
  "tasks": [
    {
      "end_date": "2018-12-12T11:33:37.490050+00:00",
      "platform_code": "winxpx86",
      "maliciousness": 100,
      "id": 16916,
      "status": "successful",
      "start_date": "2018-12-12T11:29:44.645968+00:00",
      "rules": {
        "neutral": [
          "Searching for the window",
          "Creating a window",
          "DNS request",
          "Sending an HTTP GET request"
        "suspicious": [
          "Connection attempt by exploiting the app vulnerability"
        ]
      },
      "detects": [
        "behavior",
        "files_dumps"
        ],
      "verdict": "malware2"
    },
    {
      "end_date": "2018-12-12T11:33:47.716867+00:00",
      "platform_code": "win7x86",
      "maliciousness": 100,
      "id": 16917,
      "status": "successful",
      "start_date": "2018-12-12T11:29:44.645968+00:00",
      "rules": {
```

```
        "neutral": [
          "Creating a window",
          "DNS request",
          "Sending an HTTP GET request",
          "Creating a process from a recently created file",
          "Launching a process"
        ],
        "suspicious": [
          "Connection attempt by exploiting the app vulnerability"
        ]
      },
      "detects": [
        "behavior",
        "files_dumps"
        ],
      "verdict": "malware2"
    },
    {
      "end_date": "2018-12-12T11:34:08.229276+00:00",
      "platform_code": "win7x64",
      "maliciousness": 100,
      "id": 16918,
      "status": "successful",
      "start_date": "2018-12-12T11:29:44.645968+00:00",
      "rules": {
        "neutral": [
          "Creating a window",
          "DNS request",
          "Sending an HTTP GET request",
          "Creating a file in the %temp% directory",
          "Launching a process",
          "Launching the default Windows debugger (dwwin.exe)"
        ],
        "suspicious": [
          "Connection attempt by exploiting the app vulnerability"
        ]
      },
      "detects": [
        "behavior",
        "files_dumps"
        ],
      "verdict": "malware2"
    },
    {
      "end_date": "2018-12-12T11:35:11.553665+00:00",
      "platform_code": "win10x64",
      "maliciousness": 100,
      "id": 16919,
      "status": "successful",
      "start_date": "2018-12-12T11:29:44.645968+00:00",
      "rules": {
        "neutral": [
          "Creating a window",
          "Sending an HTTP GET request"
        ],
        "suspicious": [
          "Connection attempt by exploiting the app vulnerability"
        ]
      },
      "detects": [
        "behavior",
```

```
        "files_dumps"
        ],
      "verdict": "malware2"
    },
    {
      "end_date": "2018-12-12T11:36:12.589364+00:00",
      "platform_code": "win11x64",
      "maliciousness": 100,
      "id": 16920,
      "status": "successful",
      "start_date": "2018-12-12T11:29:44.645968+00:00",
      "rules": {
        "neutral": [
          "Creating a window",
          "Sending an HTTP GET request"
        ],
        "suspicious": [
          "Connection attempt by exploiting the app vulnerability"
        ]
      },
      "detects": [
        "behavior",
        "files_dumps"
        ],
      "verdict": "malware2"
    }
  ]
}
```


Если вы запросили список анализов методом GET analyses, в ответ вы получите список объектов **Analysis**, в каждом из которых ключ `tasks` — это список объектов TaskBasic:


```
{
    "id": 1629b17b-fd44-46e6-97a2-1310c1f050a4,
    "sample_id": 6248,
    "size": 3242863,
    "sha1": "8c81eb1b6a87e32152d439965eca944bc59bdeb3",
    "start_date": "2018-12-12T11:29:44.645968+00:00",
    "user_name": "name_example",
    "format_name": "rtf",
    "tasks": [
      {
        "end_date": "2018-12-12T11:33:37.490050+00:00",
        "platform_code": "winxpx86",
        "maliciousness": 100,
        "id": 16916,
        "status": "successful",
        "start_date": "2018-12-12T11:29:44.645968+00:00"
      },
      {
        "end_date": "2018-12-12T11:33:47.716867+00:00",
        "platform_code": "win7x86",
        "maliciousness": 100,
        "id": 16917,
        "status": "successful",
        "start_date": "2018-12-12T11:29:44.645968+00:00"
      },
      {
        "end_date": "2018-12-12T11:34:08.229276+00:00",
        "platform_code": "win7x64",
        "maliciousness": 100,
```

```
        "id": 16918,
        "status": "successful",
        "start_date": "2018-12-12T11:29:44.645968+00:00"
      },
      {
        "end_date": "2018-12-12T11:35:11.553665+00:00",
        "platform_code": "win10x64",
        "maliciousness": 100,
        "id": 16919,
        "status": "successful",
        "start_date": "2018-12-12T11:29:44.645968+00:00"
      },
      {
        "end_date": "2018-12-12T11:36:12.589364+00:00",
        "platform_code": "win11x64",
        "maliciousness": 100,
        "id": 16920,
        "status": "successful",
        "start_date": "2018-12-12T11:29:44.645968+00:00"
      }
    ]
  }
```


11.4.2. APIEvent

Объект **APIEvent** содержит данные о событии, произошедшем во время выполнения файла.

Структура

Ключ	Тип	Описание
`process`	string	Полный путь к процессу на гостевой операционной системе.
`rules`	object	Список сработавших правил.
`arguments`	string	Аргументы события. Указывают на особые условия выполнения события.
`maliciousness`	integer	Вредоносность, от 0 до 100.
`event`	string	Событие, произошедшее во время выполнения файла. Соответствует общеупотребляемым API-функциям.
`timestamp`	integer	Временная метка события. Отсчитывается с момента запуска анализа файла.

Пример


```
{
  "process": "<CURRENT_DIR>\\example.exe:1432:2432",
  "rules": {
    "neutral": [
```

```
      "Connection attempt"
    ]
  },
  "arguments": "To '125.251.199.120':540",
  "maliciousness": 0,
  "event": "ConnectNet",
  "timestamp": 9
}
```

11.4.3. Call (опционально)

Объект **Call** содержит данные об исходящем телефонном звонке. Объект используется только в результатах анализа приложений для Android.

Структура

Ключ	Тип	Описание
`type`	string	Всегда `call`.
`number`	string	Телефонный номер, на который совершен звонок.

Пример

```
{
  "type": "call",
  "number": "667206"
}
```

11.4.4. Connection

Объект **Connection** содержит данные о сетевом подключении.

Структура

Ключ	Тип	Описание
`port`	integer	Номер порта.
`host`	string	Имя хоста или IP-адрес.
`country`	object	Страна.
`app`	string	Данные прикладного уровня.
`protocol`	string	Протокол, используемый для подключения.
`ip`	string	IP-адрес хоста.

Пример

```
{
  "port": 31,
  "host": "<IP-адрес>",
  "country": {
    "name": "China",
    "code3": "CHN"
  },
  "app": "{70,69,6e,67}",
  "protocol": "TCP/IP",
  "ip": "<IP-адрес>"
}
```

11.4.5. Dump

Объект **Dump** содержит данные о потенциально вредоносном дампе процесса.

Структура

Ключ	Тип	Описание
archive_path	string	Путь к файлу в архиве отчета.
name	string	Имя файла.
sha1	string	Хеш SHA1.
detect	string	Имя угрозы.
pid	integer	Идентификатор процесса.

Пример

```
{
  "archive_path": "dumps/4_89432000_a71a8d8316cb3bc.4.38.6.ndmp",
  "name": "a71a8d8316cb3bc",
  "sha1": "8f11bc1fb9ac4444472213e0ae91bc166493f0ab",
  "detect": "Trojan.Necurs.5",
  "pid": 4
}
```

11.4.6. Drop

Объект **Drop** содержит данные о файле, созданном в процессе анализа.

Структура

Ключ	Тип	Описание
`archive_path`	string	Путь к файлу в архиве отчета.
`sha1`	string	Хеш SHA1.
`detect`	string	Имя угрозы.
`path`	string	Путь к созданному файлу.

Пример

```
{
  "archive_path": "drops/d##vault.hta(0)",
  "sha1": "392b84af9ede8fc70a29f02131e9ae91ef88c809",
  "detect": "JS.DownLoader.994",
  "path": "D:\\vault.hta"
}
```

11.4.7. Format

Объект **Format** содержит данные о формате файла.

Структура

Ключ	Тип	Описание
`name`	string	Название формата файла.
`group_name`	string	Название типа файла. Возможные значения: • `apk`: пакеты Android. • `arf`: файлы Acrobat Reader. • `ja`: исполняемые файлы Java. • `js`: файлы сценарных языков. • `moc`: документы Microsoft Office. • `other`: прочие. • `uef`: исполняемые файлы *nix. • `wef`: исполняемые файлы Windows.
`platforms`	array [Platform.code]	Список платформ.

Пример

```
{
   "name": "exe",
   "group_name": "wef",
   "platforms": [
     "winxpx86",
     "win7x86",
     "win7x64",
     "win10x64",
     "win11x64"
   ]
}
```

11.4.8. Intent (опционально)

Объект **Intent** содержит данные о намерении. Объект используется только в результатах анализа приложений для Android.

Структура

Ключ	Тип	Описание
`cn`	string	Компонент, который получает намерение.
`action`	string	Название выполняемого действия.
`data`	string	Данные, с которыми выполняется действие.
`transaction`	string	Транзакция, определяющая тип запускаемого компонента: • START_ACTIVITY — запуск операции. • START_SERVICE — запуск службы. • BROADCAST_INTENT — рассылка широковещательных сообщений.
`maliciousness`	integer	Вредоносность, от 0 до 100.
`rules`	object	Список сработавших правил.
`timestamp`	integer	Временная метка. Отсчитывается с момента запуска анализа файла.

Пример

```
{
  "cn": null,
  "action": "android.app.action.ADD_DEVICE_ADMIN",
  "data": null,
  "transaction": "START_ACTIVITY",
  "maliciousness": 69,
  "rules": {
```

```
    "suspicious": [
      "Using device administration features"
    ]
  },
  "timestamp": 0
}
```

11.4.9. Message (опционально)

Объект **Message** содержит данные об исходящем SMS-сообщении. Объект используется только в результатах анализа приложений для Android.

Структура

Ключ	Тип	Описание
type	string	Всегда message.
number	string	Телефонный номер, на который отправлено сообщение.
text	string	Текст сообщения.

Пример

```
{
  "type": "message",
  "number": "000",
  "text": "Balance"
}
```

11.4.10. Platform

Объект **Platform** содержит данные о платформе ОС и в некоторых случаях — о программе для запуска файла.

Структура

Ключ	Тип	Описание
code	string	Сокращенное название платформы.
name	string	Название приложения или платформа ОС.
os_code	string	Платформа ОС.

Пример

```
{
```

```
    "code": "office_7_32",
    "name": "Microsoft Office 2010",
    "os_code": "Windows 7 32-bit"
}
```

11.4.11. Sample

Объект **Sample** содержит данные об исходном файле, загруженном на анализ.

Структура

Ключ	Тип	Описание
`id`	integer	ID файла.
`name`	string	Имя файла.
`format_name`	string	Формат файла. Определяется сервисом Dr.Web vxCube автоматически. Формат файла задает команду для выполнения файла, если она не задана явно при запуске анализа.
`is_x64`	boolean	Определяет разрядность платформы для выполнения файла. Значение `null`, если файл неисполняемый.
`md5`	string	Хеш MD5.
`sha1`	string	Хеш SHA1.
`sha256`	string	Хеш SHA256.
`size`	integer	Размер файла в байтах.
`upload_date`	string	Дата и время загрузки файла.
`platforms`	array [Platform.code]	Список поддерживаемых платформ для выполнения файла.

Пример

```
{
  "id": 42,
  "name": "sample.exe",
  "format_name": "sys",
  "is_x64": null,
  "md5": "a0b0f87193b79ac1db32f251f2f5e5b2",
  "sha1": "e54639e9d81680d0acc154d42ae7350ed481b848",
  "sha256": "51133e7e4d52b94e3360ac1866b76bf2b2bca056492bcf93de3c37d6b0c07104",
  "size": 1897856,
  "upload_date": "2018-07-31T11:42:36.873274+00:00"
  "platforms": [
    "winxpx86",
    "win7x86",
    "win7x64",
```

```
    "win10x64",
    "win11x64"
  ]
}
```

11.4.12. Session

Объект **Session** содержит данные о сессии.

Структура

Ключ	Тип	Описание
`api_key`	string	API-ключ.
`start_date`	string	Дата и время начала сессии.

Пример

```
{
    "api_key": "aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
    "start_date": "2018-12-20T08:55:35.158344+00:00"
}
```

11.4.13. Task

Объект **Task** содержит данные о задаче. Задача — это анализ файла на отдельной платформе. **Task** может содержать различный набор ключей: **TaskBasic**, **TaskFinished** или **TaskProcessing**.

TaskBasic

TaskBasic содержит общие сведения о задаче. Такой объект с базовым набором ключей используется в списке объектов Analysis.

Структура

Ключ	Тип	Описание
`id`	integer	ID задачи.
`status`	string	Статус задачи. Доступные значения: `in queue`, `failed`, `processing`, `deleted`, `successful`.
`platform_code`	string	Platform.code.

Ключ	Тип	Описание
start_date	string (datetime.iso8601)	Дата и время запуска задачи.
end_date	string/null (datetime.iso8601)	Дата и время завершения задачи.
maliciousness	integer/null	Вредоносность, от 0 до 100.

Пример

```
{
    "id": 20,
    "status": "failed",
    "platform_code": "winxpx86",
    "start_date": "2018-07-30T16:54:07.156371",
    "end_date": "2018-07-30T16:55:07.156371",
    "maliciousness": null
}
```

TaskFinished

TaskFinished содержит ключи объекта **TaskBasic** и результаты анализа файла на заданной платформе.

Структура

Ключ	Тип	Описание
detects	string[]	Список типов обнаруженных угроз. Список может включать следующие строки: yara: сработало правило YARA; behavior: обнаружено вредоносное или подозрительное поведение файла; files_dumps: найдены угрозы в созданных файлах и/или дампах памяти.
end_date	string/null (datetime.iso8601)	Дата и время окончания задачи.
id	integer	ID задачи.
maliciousness	integer/null	Вредоносность, от 0 до 100.
platform_code	string	Platform.code.

Ключ		Тип	Описание
rules		object/null	Список сработавших правил.
	malicious	string[]	Список сработавших правил вредоносной активности файла.
	neutral	string[]	Список сработавших правил нейтральной активности файла.
	suspicious	string[]	Список сработавших правил подозрительной активности файла.
sample_detect		string/null	Название угрозы, обнаруженной по вирусным базам.
start_date		string (datetime.iso8601)	Дата и время запуска задачи.
status		string	Статус задачи. Доступные значения: `in queue`, `failed`, `processing`, `deleted`, `successful`.
tags		string[]	Список тегов из сработавших правил YARA.
verdict		string	Общая оценка вредоносности файла, соответствующая одной из трех категорий. Большее цифровое значение соответствует большей вероятности вредоносности. Доступные значения: `none`, `clean1`, `clean2`, `suspicious1`, `suspicious2`, `malware1`, `malware2`.
yara_rules		object[]	Список сработавших правил YARA.
	name	string	Имя правила YARA.
	rule_type	string	Тип правила YARA. Доступные значения: `user` (пользовательское правило) и `system` (системное правило).
	severity	string	Тип вредоносности файла. Добавляя правило YARA, вы должны указать, какой тип вредоносности будет присвоен файлу при срабатывании этого правила. В поле `severity` отображается указанный вами тип. Доступные значения: `neutral`, `suspicious`, `malware`. Подробнее о том, как добавить правило YARA...

Пример

```
{
  "id": 16916,
```

```
  "status": "successful",
  "maliciousness": 100,
  "platform_code": "winxpx86",
  "start_date": "2018-12-12T11:29:44.645968+00:00",
  "end_date": "2018-12-12T11:33:37.490050+00:00",
  "verdict": "malware2",
  "rules": null,
  "detects": [
     "files_dumps"
   ],
  "platform_code": "win7x64"
}
```


TaskProcessing

TaskProcessing содержит ключи объекта **TaskBasic** и данные о ходе анализа.

Структура

Ключ	Тип	Описание
`end_date`	string	Дата и время окончания задачи.
`id`	integer	ID задачи.
`maliciousness`	integer/null	Вредоносность, от 0 до 100.
`message`	string/null	Сообщение о ходе выполнения.
`platform_code`	string	Platform.code.
`progress`	integer	Прогресс выполнения задачи, в процентах.
`start_date`	string (datetime.iso8601)	Дата и время запуска задачи.
`status`	string	Текущий статус задачи. Доступные значения: `in queue`, `failed`, `processing`, `deleted`, `successful`.

Пример


```
{
  "id": 18656,
  "status": "processing",
  "maliciousness": null,
  "platform_code": "win7x86",
  "start_date": "2019-02-07T09:39:11.517117+00:00",
  "end_date": null,
  "message": "Waiting while the file is running (60 sec)…",
  "progress": 19
}
```

11.5. Примеры

В этом разделе приведены примеры того, как работать с сервисом Dr.Web vxCube, используя API.

Вы узнаете, как:

- получить API-ключ;
- загрузить файл или архив на сервер Dr.Web vxCube;
- запустить анализ;
- получить информацию об анализе;
- скачать отчет.

11.5.1. Как получить API-ключ

Чтобы получить API-ключ, отправьте запрос POST login с логином и паролем.

Как получить API-ключ, созданный ранее

Чтобы получить один из созданных API-ключей, укажите значение параметра `new_key: false`, или просто не указывайте этот параметр:

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/login \
-H "Content-Type: application/json" \
-d "{\"login\":\"example@drweb.com\", \"password\":\"secret_password\"}"
```

Вы получите ответ с API-ключом (его требуется указывать в заголовке каждого последующего запроса):

```
{
    "new_key": false,
    "api_key": "aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
    "start_date": "2019-02-08T04:08:15.162342+00:00"
}
```

Как создать API-ключ

Чтобы создать новый API-ключ, укажите значение параметра `new_key: true` (если у вас нет созданных API-ключей, параметр можно не указывать, API-ключ будет создан все равно):

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/login \
-H "Content-Type: application/json" \
-d "{\"login\":\"example@drweb.com\", \"password\":\"secret_password\", \"new_key\":
true, \"name\":\"example_name_api\"}"
```

Вы получите ответ с API-ключом (его требуется указывать в заголовке каждого последующего запроса):

```
{
   "new_key": true,
   "api_key": "bbbbbbbb-cccc-dddd-eeee-ffffffffffff",
   "start_date": "2019-03-08T04:08:15.162342+00:00",
   "name": "example_name_api"
}
```

11.5.2. Как загрузить файл или архив на сервер vxCube

Вы можете загрузить на сервер vxCube для последующего анализа как отдельный файл, так и архив с несколькими файлами. Информация о поддерживаемых форматах файлов и архивов приведена в разделе Поддерживаемые форматы.

Как загрузить на сервер vxCube отдельный файл

Чтобы загрузить на сервер файл, отправьте запрос POST samples:

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/samples \
-F "file=@testfile.pdf" \
-F "password="vxcube"" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee"
```

В ответ вы получите объект Sample, который будет содержать данные о загруженном файле, в том числе формат файла, определенный автоматически, и список поддерживаемых платформ. Используйте данные из ответа при отправке запроса на анализ загруженного файла.

Пример ответа:

```
{
    "id": 6784,
    "size": 10881846,
    "name": "testfile.pdf",
    "is_x64": null,
    "format_name": "pdf",
    "upload_date": "2019-02-08T04:08:15.162343+00:00",
    "md5": "34fb8ae3c01653985085ee7e2f749ea5",
    "sha1": "00a610100a3516f4d0daa33e7de317d2ddb6c2c6",
    "sha256": "11bd131be00cbe1c43b4444ec4300dc7651805ea36393b1cca1675983dc275bc",
    "platforms": [
        "acrobat_xp_10",
        "acrobat_7_32_11",
        "acrobat_7_64_15",
        "acrobat_10_64_15",
        "acrobat_11_64_15"
    ]
}
```

Как загрузить на сервер vxCube архив с файлами

Чтобы загрузить на сервер архив, отправьте запрос POST samples:

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/samples \
-F "file=@testarchive.zip" \
-F "password="vxcube"" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee"
```

В ответ вы получите список объектов Sample, которые будут содержать данные о загруженных в составе архива файлах, в том числе формат файлов, определенный автоматически, и список поддерживаемых платформ. Используйте данные из ответа при отправке запроса на анализ файлов. Обратите внимание, что для каждого файла из архива потребуется отправить отдельный запрос на анализ.

Пример ответа:


```
{
    "samples": [
      {
        "id": 322277,
        "name": "script_bash.sh",
        "size": 31,
        "format_name": "sh",
        "upload_date": "2025-01-10T10:22:05.370576",
        "md5": "ee7f40fc10ebc0c5227f2307d4cc0eec",
        "sha1": "824917b5b19af65b09b4f879787bcbc304304df3",
        "sha256": "7a4b93d2a929c865da7f8fa060cf8bdeba00caa3a246c00c61819e101578c525",
        "is_x64": null,
        "platforms": [
            "intel64_astra_ce_2.12",
            "intel64_astra_se_1.7.2",
            "intel64_debian_bullseye"
        ]
      },
      {
        "id": 322278,
        "name": "Simulator.exe",
        "size": 12796104,
        "format_name": "exe",
        "upload_date": "2025-01-10T10:22:06.673229",
        "md5": "1d8a2c83aeec264d6df97f3867d13051",
        "sha1": "6e417d89c4a6ae436f815ee038255f5dbf31c1ca",
        "sha256": "b1fd18441460ed1bd9b6a8107f2f09ddc971ed33ddbca53c6a38124f6830b2d9",
        "is_x64": false,
        "platforms": [
            "win11x64",
            "win10x64",
            "win7x64",
            "win7x86",
            "winxpx86"
        ]
      },
      {
        "id": 322279,
        "name": "Welcome.doc",
        "size": 28909,
        "format_name": "odt",
```

```
        "upload_date": "2025-01-10T10:22:06.684141",
        "md5": "64df4748a0e674cb65601a1157a1c900",
        "sha1": "0bfac9df80cade5b8b03576194549b197fc34388",
        "sha256": "7ce72f1ea01bea3d99e4658e5dc909e662e10c65d83d280aa4bf0e25526bf752",
        "is_x64": null,
        "platforms": [
            "office_11_64",
            "office_10_64",
            "office_7_32",
            "office_7_64",
            "office_xp"
        ]
      }
    ]
}
```


11.5.3. Как запустить анализ

После того как вы загрузите файл на сервер vxCube, вы сможете запустить анализ файла. Для этого отправьте запрос POST analyses, указав ID загруженного файла и список платформ, на которых нужно выполнить файл. Значения параметров берутся из ответа, полученного на API-запрос загрузки файла.

Пример запроса:

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/analyses \
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee" \
-d "{\"sample_id\":\"6784\", \"platforms\":[\"acrobat_7_32_11\", \"acrobat_7_64_15\"]}"
```

Чтобы запустить анализ с перенаправлением сетевого трафика, отправьте следующий запрос POST analyses:

```
curl -X POST https://<IP-адрес/доменное имя сервера>/api-2.0/analyses \
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee" \
-d "{\"sample_id\":\"6784\", \"platforms\":[\"acrobat_7_32_11\", \"acrobat_7_64_15\",
\"net\": \"socks5://username:password@<proxyaddress>:1080?udp=on\"}"]}"
```

В ответ вы получите объект Analysis, который содержит общие сведения об анализе:


```
{
    "id": 6260,
    "sample_id": 6784,
    "size": 10881846,
    "sha1": "00a610100a3516f4d0daa33e7de317d2ddb6c2c6",
    "start_date": "2019-02-08T04:08:15.162343+00:00",
    "format_name": "pdf",
    "user_name": "example@drweb.com",
    "tasks": [{
        "message": null,
        "end_date": null,
        "platform_code": "acrobat_7_64_15",
        "maliciousness": null,
        "progress": 0,
        "id": 18676,
        "status": "in queue",
        "start_date": "2019-02-08T04:08:15.643122+00:00"
```

```
    }, {
        "message": null,
        "end_date": null,
        "platform_code": "acrobat_7_32_11",
        "maliciousness": null,
        "progress": 0,
        "id": 18675,
        "status": "in queue",
        "start_date": "2019-02-08T04:08:15.632924+00:00"
    }]
}
```

11.5.4. Как получить информацию об анализе

Чтобы получить подробную информацию об анализе, дождитесь его завершения и отправьте запрос GET analyses/<analysis_id:uuid>. В запросе укажите ID анализа:

```
curl -X GET https://<IP-адрес/доменное имя сервера>/api-2.0/analyses/60e21c98-7c2a-4112-
81b5-a577f6cdf4db \
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee"
```

В ответ вы получите объект Analysis:

```
{
    "id": "111ba12c5-d330-40eb-b988-fa16402ee111",
    "sha1": "9e92e9408afdf75fc3dea5e457cb0c70728f74ce",
    "sample_id": 77236,
    "size": 156160,
    "format_name": "dll",
    "start_date": "2024-02-13T14:28:08.871359",
    "user_name": "test@test.com",
    "tasks": [
        {
            "id": 235182,
            "status": "successful",
            "platform_code": "win10x64",
            "start_date": "2024-02-13T14:28:09.135345",
            "end_date": "2024-02-13T14:30:46.776797",
            "maliciousness": 94,
            "verdict": "malware2",
            "detects": [
                "yara"
            ],
            "sample_detect": null,
            "rules": {
                "neutral": [
                    "Creating synchronization primitives",
                    "Searching for synchronization primitives"
                ]
            },
            "yara_rules": [
                {
                    "name": "gozi3",
                    "severity": "malware",
                    "rule_type": "system"
                },
                {
                    "name": "gozi",
```

```
                    "severity": "malware",
                    "rule_type": "system"
                }
            ],
            "tags": [
                "GOZI3",
                "GOZI"
            ]
        }
    ]
}
```

11.5.5. Как скачать отчет

Чтобы скачать архив отчета об анализе целиком, отправьте запрос GET analyses/<analysis_id:uuid>/archive:

```
curl -X GET https://<IP-адрес/доменное имя сервера>/api-2.0/analyses/40e2fc98-1c2a-4112-
81b5-a57df2cd22db/archive \
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee" \
-o <output_archive>
```

Чтобы скачать один из файлов отчета, отправьте запрос GET tasks/<task_id:number>/archive_storage. Пример запроса на скачивание файла PCAP:

```
curl -X GET https://<IP-адрес/доменное имя сервера>/api-2.0/tasks/18681/archive_storage
\
-H "Content-Type: application/json" \
-H "Authorization: api-key aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee" \
-d "{\"path\": \"network.pcap\"}" \
-o some_file
```

12. Управление пользователями

Чтобы открыть страницу с пользователями

- В верхней части главной страницы Dr.Web vxCube выберите **Пользователи**.

Учетные данные пользователей представлены в виде таблицы. Для каждого пользователя отображается следующая информация:

- логин пользователя,
- тип учетной записи (администратор или пользователь),
- статус (активен или заблокирован),
- количество загруженных файлов,
- дата добавления пользователя.

Чтобы задать количество записей, отображаемых на одной странице

- Нажмите выпадающее меню под таблицей.

Чтобы отсортировать записи по любому из столбцов

- Нажмите заголовок соответствующего столбца.

Для сортировки записей в обратном порядке, нажмите заголовок столбца еще раз.

Чтобы отфильтровать записи

- Введите строку в поле поиска. Поиск осуществляется по всем столбцам таблицы.

Дополнительные возможности

Пользователь с правами администратора может добавить, удалить, отредактировать, заблокировать или разблокировать пользователя.

Чтобы добавить нового пользователя

1. На странице **Пользователи** выберите **Добавить**.
2. В открывшемся окне введите данные пользователя и нажмите **Добавить**.

Чтобы отредактировать учетные данные пользователя

1. Для соответствующего пользователя наведите курсор на значок ••• и выберите **Редактировать**.
2. Внесите необходимые изменения и нажмите **Применить**.

Чтобы удалить пользователя

1. Для соответствующего пользователя наведите курсор на значок ••• и выберите **Удалить**.
2. В появившемся диалоге подтвердите удаление.

Чтобы заблокировать пользователя

1. Для соответствующего пользователя наведите курсор на значок ••• и выберите **Заблокировать**.
2. В появившемся диалоге подтвердите блокировку.

Чтобы разблокировать пользователя

1. Для соответствующего пользователя наведите курсор на значок ••• и выберите **Разблокировать**.
2. В появившемся диалоге подтвердите разблокировку.

13. Как удалить Dr.Web vxCube

Для очистки сервера от компонентов в случае ошибок установки используется специальный скрипт. Его нужно запускать на каждом узле, на котором установлены компоненты Dr.Web vxCube.

Если скрипт очистки применить на хосте с `vxcube_web_host`, то он удалит базу данных службы vxcube-web, включая результаты предыдущих проверок, созданных пользователей и все прочие данные.

Очистка с сохранением виртуальных машин Windows

Чтобы сократить время переустановки, укажите аргумент `keepvm` при запуске скрипта. Это сохранит информацию об уже клонированных виртуальных машинах на узлах с Windows-анализаторами (`hyperbox_hosts`).

В таком случае вы сможете перезапустить инсталлятор с опцией hyperbox_hbsetup: false (флаг необходимости переклонирования виртуальных машин) и не клонировать машины заново.

Запуск скрипта

Чтобы запустить скрипт

1. Сохраните содержимое ниже в файл `clean.sh`.
2. Для запуска выполните команду `sudo bash clean.sh`.

```
clear

echo 'Внимание! Этот скрипт очистки полностью удалит все данные и
сервисы, связанные с установкой vxCube, включая базу данных,
результаты проверок, созданных пользователей, настройки интеграции с
другими сервисами и т. д.'

while true; do

    read -p "Продолжить очистку? (yes/no) " yn

    case $yn in

        [Yy]* ) echo go clean; break;;

        [Nn]* ) exit;;

        * ) echo "Нужен ответ Yes или No.";;

    esac
```

```
done


set -x


# docker services
docker container stop $(docker container ls -aq)
docker system prune -a --volumes -f
docker image prune -a -f
systemctl stop docker
rm -rf /etc/docker
apt purge -y docker-ce
apt purge -y docker.io
rm -rf /etc/systemd/system/multi-
user.target.wants/containerd.service
# docker vxcube web
rm -rf /var/lib/vxcube
rm -rf /opt/vxcube
rm -rf /var/log/vxcube
userdel -f -r vxcube
# docker yara
rm -rf /etc/yara_service /var/log/yara_service
# docker drweb
rm -rf /etc/drweb-service /var/log/drweb


# vxcube-flow-api
systemctl stop vxcube-flow-api
systemctl disable vxcube-flow-api
rm -rf /var/lib/vxcube-flow-api /var/log/vxcube-flow-
api /etc/vxcube-flow-api
rm -rf /etc/apt/sources.list.d/*
rm -rf /etc/systemd/system/vxcube-flow-api.service
rm -rf /etc/systemd/system/multi-user.target.wants/vxcube-flow-
api.service
userdel -f -r hyperbox-api
```

```
# linuxbox
systemctl stop linuxbox-routes
systemctl stop linuxbox_rpc
systemctl disable linuxbox-routes
systemctl disable linuxbox_rpc
rm -rf /etc/systemd/system/linuxbox-routes.service
rm -rf /etc/systemd/system/linuxbox_rpc.service
/var/lib/linuxbox/routes_reset.sh
rm -rf /var/lib/linuxbox
rm -rf /var/lib/storage/linuxbox-*
rm -rf /etc/linuxbox
apt purge -y qemu*

# dimas
systemctl stop dimas_android7.1_vxcube
systemctl stop dimasnet
systemctl disable dimas_android7.1_vxcube
systemctl disable vboxapi_android
rm -rf /etc/systemd/system/dimas*.service
rm -rf /etc/systemd/system/vboxapi_android.service
rm -rf /etc/systemd/system/multi-
user.target.wants/apkrobot_*.service /etc/systemd/system/apkrobot_*.
service
userdel -f -r dimas
rm -rf /var/lib/dimas
rm -rf /var/log/dimas
rm -rf /etc/dimas

# hyperbox
systemctl stop hbcheck
systemctl stop vboxsvc
systemctl stop hyperbox_winxpx86_vxcube hyperbox_win7x64_vxcube
hyperbox_win7x86_vxcube hyperbox_win10x64_1903_vxcube
hyperbox_win10x64_1511_vxcube
```

```
systemctl disable vboxsvc hbcheck hyperbox_winxpx86_vxcube
hyperbox_win7x64_vxcube hyperbox_win7x86_vxcube
hyperbox_win10x64_1903_vxcube hyperbox_win10x64_1511_vxcube

systemctl disable vboxdrv vboxautostart-service vboxballoonctrl-
service

rm -rf /etc/systemd/system/hbcheck.service

rm -rf /etc/systemd/system/hyperbox_*.service

rm -rf /etc/systemd/system/vboxapi.service

rm -rf /etc/systemd/system/vboxnet.service

rm -rf /etc/systemd/system/vboxsvc.service

rm -
rf /etc/fakenet /etc/vbox /etc/hyperbox /var/lib/vboxnet_workspace

rm -rf /var/log/hyperbox /var/log/vbox*

if [ "$1" == "keepvm" ]; then

  # this will keep vms and configs

  apt remove -y virtualbox-hyperbox

else

  # this will delete all

  apt purge -y virtualbox-hyperbox

  userdel -f -r hyperbox

  rm -rf /var/lib/hyperbox

  rm /var/lib/storage/* -r

fi

apt purge drweb-procdump -y

apt purge aksusbd -y


# evparser

systemctl stop evparser

systemctl disable evparser

rm -rf /var/lib/evparser /var/lib/evparser/
.cache /etc/evparser /var/log/evparser

rm -rf /etc/systemd/system/evparser.service

rm -rf /etc/systemd/system/multi-user.target.wants/evparser.service

userdel -f -r evparser


# pogreb
```

```
rm -rf /etc/pogreb-client /var/log/pogreb-client

# ftp
systemctl stop proftpd
sudo apt purge -y proftpd*
rm -rf /etc/proftpd /var/log/proftpd
rm -rf /srv/vxcube

# dhcp
apt purge -y dnsmasq
apt purge -y isc-dhcp-server
# remove includes from dhcp config (they will not be deleted by
dpkg)
DHCP_CONF=/etc/dhcp/dhcpd.conf
if [ -f $DHCP_CONF ] ; then
    sed -i 's#include "/etc/dhcp/dhcpd.vbox";##g' $DHCP_CONF
    sed -i 's#include "/etc/dhcp/dhcpd_android.vbox";##g' $DHCP_CONF
fi
rm /etc/dhcp/dhcpd.vbox
rm /etc/dhcp/dhcpd_android.vbox

# openvpn
systemctl stop openvpn
rm -rf /var/log/openvpn
rm -rf /etc/openvpn
apt purge -y openvpn

# nginx
systemctl stop nginx
# dpkg warns on non-empty
rm -rf /etc/nginx /var/www/html /var/log/nginx
apt purge -y nginx*

# rabbitmq
```

```
systemctl stop rabbitmq-server
apt purge -y rabbitmq-server

# zabbix
apt purge -y zabbix-*
rm -rf /etc/systemd/system/multi-user.target.wants/zabbix-
agent.service

# all virtualenvs
rm -rf /var/lib/virtualenvs

apt purge -y python-pip
apt purge -y python3-pip

# firewall clean
iptables -t nat -F
iptables -t mangle -F
iptables -F
iptables -X
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
iptables-save | sudo tee /etc/iptables/rules.v4 >> /dev/null

apt autoremove -y

systemctl daemon-reload
systemctl reset-failed
```

14. Техническая поддержка

При возникновении проблем с работой Dr.Web vxCube вы можете связаться со службой технической поддержки «Доктор Веб» следующими способами:

- Заполните веб-форму: https://support.drweb.com/support_wizard/vxcube.
- Позвоните по телефону в Москве: +7 (495) 789-45-86 или по бесплатной линии для всей России: 8-800-333-7932.

Данные о региональных представительствах и офисах компании «Доктор Веб» вы можете найти на официальном сайте по адресу https://company.drweb.com/contacts/offices/.

15. Приложение А. Список программного обеспечения на виртуальных машинах

Windows XP x86

- Microsoft Office Enterprise 2007 x86 (опционально)
- Adobe Acrobat Reader 10.1.0
- Adobe Flash 12.0.0.77
- JAVA 6u45
- Adobe Flash Standalone 10.3.181.23 (%windir%\flash_sa.exe)
- Mozilla Firefox 52.0.2
- Opera 35.0
- Google Chrome 44.0.2403.155
- ICQ 8.3 build 7317
- QIP 2012 4.0.9380
- Pidgin 2.10.11
- Mozilla Thunderbird 31.7.0
- Visual C++ Redistributable 2005 x86
- Visual C++ Redistributable 2008 x86
- Visual C++ Redistributable 2010 x86
- Visual C++ Redistributable 2012 x86
- Visual C++ Redistributable 2013 x86
- Visual C++ Redistributable 2015 x86
- .NET Framework 1.1
- .NET Framework SDK (msvcp70.dll, msvcr70.dll)
- .NET Framework 2.0 Service Pack SP2
- .NET Framework 3.0 Service Pack SP2
- .NET Framework 3.5 Service Pack SP1
- .NET Framework 4.0
- Steam 2.91
- WinRAR 5.20 x86
- Telegram Desktop 1.2.17
- mIRC 7.43

Windows 7 x86

- Adobe Acrobat Reader 11.0.1
- Microsoft Office Professional Plus 2010 x86 (опционально)

- Adobe Flash 12.0.0.77
- Adobe Flash ActiveX 17.0.0.188
- JAVA 7u11
- Adobe Flash Standalone 11.1.102.62 (%windir%\flash_sa.exe)
- Mozilla Firefox 68.0.2
- Opera 33.0.1990.115
- Google Chrome 43.0.2357.65
- ICQ 8.3 build 7317
- QIP 2012 4.0.9380
- Pidgin 2.10.11
- Mozilla Thunderbird 31.7.0я
- Visual C++ Redistributable 2005 x86
- Visual C++ Redistributable 2008 x86
- Visual C++ Redistributable 2010 x86
- Visual C++ Redistributable 2012 x86
- Visual C++ Redistributable 2013 x86
- Visual C++ Redistributable 2015 x86
- .NET Framework 1.1
- .NET Framework SDK (msvcp70.dll, msvcr70.dll)
- .NET Framework 4.8
- Steam 3.17
- .NET Framework 4.7.1
- Telegram Desktop 1.2.17
- WinRAR 5.20 x86

Windows 7 x64

- Adobe Acrobat Reader Document Cloud 15.10.20056
- Microsoft Office Professional Plus 2010 x64 (опционально)
- Adobe Flash 18.0.0.261
- Adobe Flash ActiveX 19.0.0.207
- JAVA 8u45 x64
- Adobe Flash Standalone 19.0.0.226 (%windir%\flash_sa.exe)
- K-Lite Mega Codec Pack 11.1.0
- Mozilla Firefox 78.0.2
- Opera 29.0.1795.47
- Google Chrome 42.0.2311.135

- ICQ 8.3 build 7317
- Mail.Ru Agent 6.4 build 8614
- QIP 2012 4.0.9380
- Pidgin 2.10.11
- Total Commander 8.51a x64
- Mozilla Thunderbird 31.6.0
- Winamp 5.666
- Visual C++ Redistributable 2005 x86
- Visual C++ Redistributable 2008 x86
- Visual C++ Redistributable 2010 x86
- Visual C++ Redistributable 2012 x86
- Visual C++ Redistributable 2013 x86
- Visual C++ Redistributable 2015 x86
- Visual C++ Redistributable 2005 x64
- Visual C++ Redistributable 2008 x64
- Visual C++ Redistributable 2010 x64
- Visual C++ Redistributable 2012 x64
- Visual C++ Redistributable 2013 x64
- Visual C++ Redistributable 2015 x64
- .NET Framework 1.1
- .NET Framework SDK (msvcp70.dll, msvcr70.dll)
- .NET Framework 4.8
- Steam 3.17
- Telegram Desktop 1.4.3
- .NET Framework 4.7.1
- WinRAR 5.3 x64
- mIRC 7.41

Windows 10 x64

- Adobe Acrobat Reader Document Cloud 2015.010.20060
- Adobe Flash 21.0.0.197
- Adobe Flash ActiveX 21.0.0.197
- Microsoft Office Professional Plus 2016 x86 (опционально)
- JAVA 8u77 x64
- Adobe Flash Standalone 19.0.0.226 (%windir%\flash_sa.exe)
- Mozilla Firefox 91.0.2 x64

- Opera 36.0.2130.46
- Google Chrome 47.0.2526.80
- ICQ 8.3 build 7317
- QIP 2012 4.0.9380
- Pidgin 2.10.11
- Mozilla Thunderbird 38.7.1
- Visual C++ Redistributable 2005 x86
- Visual C++ Redistributable 2008 x86
- Visual C++ Redistributable 2010 x86
- Visual C++ Redistributable 2012 x86
- Visual C++ Redistributable 2013 x86
- Visual C++ Redistributable 2017 x86
- Visual C++ Redistributable 2005 x64
- Visual C++ Redistributable 2010 x64
- Visual C++ Redistributable 2012 x64
- Visual C++ Redistributable 2013 x64
- Visual C++ Redistributable 2017 x64
- .NET Framework 4.6.2
- Steam 3.37
- Telegram Desktop 1.4.3
- mIRC 7.43
- WinRAR 5.31 x64

Windows 11 x64

- Adobe Acrobat Reader Document Cloud 2015.010.20060
- Microsoft Office Professional Plus 2016 x86 (опционально) 16.0.4266.1001
- JAVA 8u77 x64
- Mozilla Firefox 91.0.2 x64
- Opera 36.0.2130.46
- Google Chrome 47.0.2526.106
- Mozilla Thunderbird 78.9.1 x64
- Visual C++ Redistributable 2008 x86
- Visual C++ Redistributable 2010 x86
- Visual C++ Redistributable 2012 x86
- Visual C++ Redistributable 2013 x86
- Visual C++ Redistributable 2015-2019 x86

- Visual C++ Redistributable 2010 x64
- Visual C++ Redistributable 2012 x64
- Visual C++ Redistributable 2013 x64
- Visual C++ Redistributable 2015-2019 x64
- Steam 2.10.91.91
- Telegram Desktop 1.4.3
- WinRAR 5.31.0 x64

Android 7.1

- Android Keyboard (AOSP) 7.1.2
- Calculator 7.1.2
- Calendar 7.1.2
- Camera 2.0.002
- Clock 4.5.0
- Contacts 1.4.22
- Dev Tools 1.0
- Email 7.1.2
- Files 7.1.2
- Gallery 1.1.40030
- Google Play 31.6.13-21
- Google Play Games 2022.01.32371
- Google Play Services 22.09.20
- Launcher3 7.1.2
- Messaging 1.0.001
- Music 3.0
- NotePad 7.1.2
- Phone 3.00.00
- RSS Reader 7.1.2
- Search 7.1.2
- Settings 7.1.2
- Terminal Emulator 1.0.70
- WebView Shell 1.0

Astra SE 1.7 (Воронеж)

- Стандартный пакет программного обеспечения

Astra CE 2.12 (Орел)

- Стандартный пакет программного обеспечения

Debian 8 (Jessie) ARMel 32-bit

- Стандартный пакет программного обеспечения

Debian 8 (Jessie) PowerPC 32-bit

- Стандартный пакет программного обеспечения

Debian 10 (Buster) MIPS 32-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) ARM 64-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) ARMhf 32-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) Intel 32-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) Intel 64-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) MIPSel 32-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) MIPSel 64-bit

- Стандартный пакет программного обеспечения

Debian 11 (Bullseye) PowerPCel 64-bit

- Стандартный пакет программного обеспечения

16. Приложение Б. Функции модуля dr_sandbox

Функции для песочницы Android (категория andr)

archive_files

certificate_sha1

dynamic

created_files

path
sha1

crypto_dumps

downloaders

detect
sha1

downloads

detect
sha1
url

droppers

detect
sha1

dumps

detect
path
sha1

executed_commands

flags

phone_calls

sms

message

number

urls

manifest

activities

app_name

filters

home_activity

is_firmware

main_activity

meta_data

name

resource

value

package

permissions

receivers

services

strings_resources

version_code

version_name

resources_digests

sha1

Загружает перечисленные файлы (download_file)

Загружает файлы (download_files)

Запускает на исполнение (exec)

Выполняет операции WMI (exec_wmi)

Создает и запускает на исполнение (эксплойт) (exploit_create_and_exec)

Создает и загружает библиотеки (эксплойт) (exploit_create_and_load_library)

Запускает на исполнение (эксплойт) (exploit_exec)

Разрешает автозапуск со съемных носителей (force_autorun_for_removable_media)

Блокирует отображение расширений файлов (hide_from_view_file_extensions)

Блокирует отображение скрытых файлов (hide_from_view_hidden_files)

Скрывает процессы (hide_processes)

Отключает уведомления панели задач (hide_taskbar_notifications)

Перехватывает функции в браузерах (hook_in_browser)

Устанавливает перехват сообщений о нажатии для всех процессов (hook_keyboard_all_processes)

Устанавливает перехват сообщений о нажатии для перечисленных процессов (hook_keyboard_concrete_processes)

Устанавливает перехват оконных сообщений (hook_keyboard_on_window_messages)

Внедряет код в большое количество пользовательских процессов (inject_to_a_lot_of_user_processes)

Внедряет код в системные процессы (inject_to_system_proc)

Внедряет код в пользовательские процессы (inject_to_user_proc)

Изменяет настройки проводника Windows (modify_explorer_settings)

Изменяет настройки браузера Windows IE (modify_ie_settings)

Удаляет или модифицирует реестр (modify_registry_to_bypass_firewall)

Изменяет DNS-сервер (modify_system_dns)

Изменяет системные настройки (modify_system_settings)

Читает файлы, отвечающие за хранение паролей (read_third_party_passwords)

Регистрирует BHO (register_bho)

Регистрирует COM-сервер (register_com_server)

Регистрирует фильтр файловой системы (register_filesystem_filter)

Устраняет перехваты функций в SSDT (restore_ssdt_hooks)

Ищет ветки реестра, отвечающие за хранение паролей (search_password_in_registry)

Ищет окна для обнаружения утилит для анализа (search_wnd_for_analyzing_soft)

Ищет окна для обнаружения программ и игр (search_wnd_for_programs_and_games)

Ищет окна для обхода антивирусов (search_wnd_to_bypass_av)

Ищет окна для обхода системы защиты файлов Windows (search_wnd_to_bypass_wfp)

Перехватывает функции в SSDT (set_concrete_ssdt_hooks)

Устанавливает стартовую страницу для браузера Windows IE (set_homepage_for_ie)

Перехватывает функции в SSDT (set_ssdt_hooks)

Завершает большое количество пользовательских процессов (try_to_terminate_a_lot_of_user_processes)

Завершает системные процессы (try_to_terminate_system_processes)

Завершает пользовательские процессы (try_to_terminate_user_processes)

Другое (категория miscellaneous)

Добавляет корневой сертификат (add_root_certificate)

Создает и запускает на исполнение (create_and_exec)

Отключает сертификат (disable_certificate)

Запускает на исполнение (exec)

Загружает драйверы (load_driver)

Изменяет значение AutoConfigURL на указанное (modify_auto_config_url)

Удаляет каталоги (remove_dir)

Удаляет файлы (remove_file)

Демонтирует файловые системы (unmount_file_system)

Вредоносные функции (категория malicious)

Пытается завершить системные процессы (attempt_kill_system_proc)

Пытается завершить приложения-анализаторы (attept_kill_analyzers)

Пытается завершить процессы (attept_kill_proc)

Компилирует исходный код (compile_program_from_source_codes)

Получает права суперпользователя (root) (gain_root_privileges)

Получает доступ к ключам SSH (get_access_to_ssh_keys)

Встраивается в процессы (inject_to_proc)

Завершает приложения-анализаторы (kill_analyzers)

Завершает процессы (kill_proc)

Завершает системные процессы (kill_system_proc)

Запускает себя как управляющую программу (launch_itself_as_daemon)

Запускает процессы (launch_processes)

Управляет службами (manage_services)

Изменяет настройки брандмауэра (modify_firewall_settings)

Изменяет настройки маршрутизатора (modify_router_settings)

Использует модули ядра (operate_kernel_modules)

Отслеживает процессы (perform_process_tracing)

Самоудаляется (remove_self)

Удаляет системные файлы (remove_system_files)

Заменяет системные файлы (replace_system_files)

Останавливает системные службы (stops_system_services)

Подменяет имя приложения (substitute_application_name_for)

Сетевая активность (категория network)

Проводит атаку перебором по SSH (attack_bruteforce_via_ssh)

Проводит атаку перебором по TELNET (attack_bruteforce_via_telnet)

Проводит атаку перебором по неизвестному протоколу (attack_bruteforce_via_unk_protocol)

Подключается к серверам (connect_to)

Подключается к серверам по протоколу IRC (connect_to_irc)

DNS-запросы (dns_ask)

Запросы HTTP GET (http_get)

Другие запросы HTTP (http_other)

Запросы HTTP POST (http_post)

Ждет входящие подключения на портах (listening_port)

Получает данные с серверов (receive_data_from_server)

Отправляет данные на серверы (send_data_to_server)

Другое (категория other)

Собирает информацию о ЦП (collect_cpu_info)

Собирает информацию о сетевой активности (collect_network_info)

Собирает информацию об ОС (collect_os_info)

Собирает информацию о RAM (collect_ram_info)

Читает информацию из /proc/kallsyms (read_info_from_proc_kallsyms)

Детекты (категория detects)

Все детекты (all_detects_here)

- Детекты файлов alloc (detects_of_allocs)
- Детекты дропов (detects_of_drops)
- Детекты дампов (detects_of_dumps)
- Детекты инжектов (detects_of_injects)
- Детекты файлов src (detects_of_src)

Проверяет буфер по смещению (check_buffer)

Проверяет байт по смещению (check_byte)

Проверяет DWORD по смещению (check_dword)

Проверяет WORD по смещению (check_word)

Ищет нечувствительную к регистру строку ASCII/wide (ci_any)

Ищет нечувствительную к регистру строку ASCII (ci_ascii)

Ищет нечувствительную к регистру строку wide (ci_wide)

Ищет нечувствительную к регистру XOR-строку (ci_xor)

Вычисляет хеш crc32 для буфера (crc32)

Ищет чувствительную к регистру строку ASCII/wide (cs_any)

Ищет чувствительную к регистру строку ASCII (cs_ascii)

Ищет чувствительную к регистру строку wide (cs_wide)

Возвращает детекты для файла (detects_of_this_file)

Ищет имя файла (filename)

Ищет имя файла с boost::regex (filename_boost_regex)

Ищет операции файловой системы (filesystem_access)

Ищет сетевые операции (network_access)

Ищет операции с реестром (registry_access)

Возвращает тип файла (sb_filetype)

Ищет подстроку в буфере (search_substring_in_range)

Описание функций для песочницы Android (категория andr)

Функция	Результат	Примеры
archive_file(regex)	Список файлов в архиве APK, соответствующих паттернам ARCHIVE_FILES_PATTERN = ['.dll', '.js', '.html', '.so'].	dr_sandbox.andr.archive_files(/pattern/)
archive_file_num	Список файлов в архиве APK, соответствующих паттернам ARCHIVE_FILES_PATTERN = ['.dll', '.js', '.html', '.so'].	dr_sandbox.andr.archive_files_num
certificate_sha1(regex)	Хеш SHA1 сертификата, которым подписано приложение.	dr_sandbox.andr.certificate_sha1(/pattern/)
certificate_sha1_num	Хеш SHA1 сертификата, которым подписано приложение.	dr_sandbox.andr.certificate_sha1_num
Подкатегория **dynamic**		
created_files.path(regex)	Созданные файлы: путь.	dr_sandbox.andr.**dynamic**.created_files.path(/pattern/)
created_files.path_num	Созданные файлы: путь.	dr_sandbox.andr.**dynamic**.created_files.path_num
created_files.sha1(regex)	Созданные файлы: SHA1.	dr_sandbox.andr.**dynamic**.created_files.sha1(/pattern/)
created_files.sha1_num	Созданные файлы: SHA1.	dr_sandbox.andr.**dynamic**.created_files.sha1_num
crypto_dumps(regex)	Шифрованные дампы.	dr_sandbox.andr.**dynamic**.crypto_dumps(/pattern/)
crypto_dumps_num	Шифрованные дампы.	dr_sandbox.andr.**dynamic**.crypto_dumps_num
downloaders.detect(regex)	Список семплов, которые скачивают анализируемый семпл.	dr_sandbox.andr.**dynamic**.downloaders.detect(/pattern/)
downloaders.detect_num	Список семплов, которые скачивают	dr_sandbox.andr.**dynamic**.downloaders.detect_num

Функция	Результат	Примеры
	анализируемый семпл.	
downloaders.sha1(regex)	Список семплов, которые скачивают анализируемый семпл.	dr_sandbox.andr.**dynamic**.downloaders.sha1(/pattern/)
downloaders.sha1_num	Список семплов, которые скачивают анализируемый семпл.	dr_sandbox.andr.**dynamic**.downloaders.sha1_num
downloads.detect(regex)	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.detect(/pattern/)
downloads.detect_num	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.detect_num
downloads.sha1(regex)	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.sha1(/pattern/)
downloads.sha1_num	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.sha1_num
downloads.url(regex)	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.url(/pattern/)
downloads.url_num	Скачанные полезные данные (apk/dex).	dr_sandbox.andr.**dynamic**.downloads.url_num
droppers.detect(regex)	Список семплов, которые загружают анализируемый семпл.	dr_sandbox.andr.**dynamic**.droppers.detect(/pattern/)
droppers.detect_num	Список семплов, которые загружают анализируемый семпл.	dr_sandbox.andr.**dynamic**.droppers.detect_num
droppers.sha1(regex)	Список семплов, которые загружают анализируемый семпл.	dr_sandbox.andr.**dynamic**.droppers.sha1(/pattern/)
droppers.sha1_num	Список семплов, которые загружают анализируемый семпл.	dr_sandbox.andr.**dynamic**.droppers.sha1_num
dumps.detect(regex)	Дамп полезных данных: детект.	dr_sandbox.andr.**dynamic**.dumps.detect(/pattern/)

Функция	Результат	Примеры
dumps.detect_num	Дамп полезных данных: детект.	dr_sandbox.andr.**dynamic**.dumps.detect_num
dumps.path(regex)	Дамп полезных данных: путь.	dr_sandbox.andr.**dynamic**.dumps.path(/pattern/)
dumps.path_num	Дамп полезных данных: путь.	dr_sandbox.andr.**dynamic**.dumps.path_num
dumps.sha1(regex)	Дамп полезных данных: хеш SHA1.	dr_sandbox.andr.**dynamic**.dumps.sha1(/pattern/)
dumps.sha1_num	Дамп полезных данных: хеш SHA1.	dr_sandbox.andr.**dynamic**.dumps.sha1_num
executed_commands(regex)	Запущенные shell-команды.	dr_sandbox.andr.**dynamic**.executed_commands(/pattern/)
executed_commands_num	Запущенные shell-команды.	dr_sandbox.andr.**dynamic**.executed_commands_num
flags(regex)	Флаги поведения.	dr_sandbox.andr.**dynamic**.flags(/pattern/)
flags_num	Флаги поведения.	dr_sandbox.andr.**dynamic**.flags_num
phone_calls(regex)	Совершенные телефонные вызовы.	dr_sandbox.andr.**dynamic**.phone_calls(/pattern/)
phone_calls_num	Совершенные телефонные вызовы.	dr_sandbox.andr.**dynamic**.phone_calls_num
sms.message(regex)	Отправленные СМС: текст сообщения.	dr_sandbox.andr.**dynamic**.sms.message(/pattern/)
sms.message_num	Отправленные СМС: текст сообщения.	dr_sandbox.andr.**dynamic**.sms.message_num
sms.number(regex)	Отправленные СМС: телефонный номер.	dr_sandbox.andr.**dynamic**.sms.number(/pattern/)
sms.number_num	Отправленные СМС: телефонный номер.	dr_sandbox.andr.**dynamic**.sms.number_num
urls(regex)	Найденные URL-адреса. Учитываются только адреса,	dr_sandbox.andr.**dynamic**.urls(/pattern/)

Функция	Результат	Примеры
	удовлетворяющие регулярному выражению.	
urls_num	Найденные URL-адреса.	dr_sandbox.andr.**dynamic**.urls_num
Подкатегория **manifest**		
activities(regex)	Список активностей (экранов) приложения.	dr_sandbox.andr.**manifest**.activities(/pattern/)
activities_num	Список всех активностей (экранов) приложения.	dr_sandbox.andr.**manifest**.activities_num
app_name(regex)	Имя приложения на устройстве.	dr_sandbox.andr.**manifest**.app_name(/pattern/)
app_name_num	Имя приложения на устройстве.	dr_sandbox.andr.**manifest**.app_name_num
filters(regex)	Список действий из манифеста.	dr_sandbox.andr.**manifest**.filters(/pattern/)
filters_num	Список действий из манифеста.	dr_sandbox.andr.**manifest**.filters_num
home_activity(regex)	Активность, точка входа в приложение.	dr_sandbox.andr.**manifest**.home_activity(/pattern/)
home_activity_num	Активность, точка входа в приложение.	dr_sandbox.andr.**manifest**.home_activity_num
is_firmware(regex)	Приложение из прошивки или нет.	dr_sandbox.andr.**manifest**.is_firmware(/pattern/)
is_firmware_num	Приложение из прошивки или нет.	dr_sandbox.andr.**manifest**.is_firmware_num
main_activity(regex)	Главная активность, точка входа в приложение.	dr_sandbox.andr.**manifest**.main_activity(/pattern/)
main_activity_num	Главная активность, точка входа в приложение.	dr_sandbox.andr.**manifest**.main_activity_num
meta_data.name(regex)	Метаданные: имя.	dr_sandbox.andr.**manifest**.meta_data.name(/pattern/)
meta_data.name_num	Метаданные: имя.	dr_sandbox.andr.**manifest**.meta_data.name_num

Функция	Результат	Примеры
meta_data.resource(regex)	Метаданные: ресурс.	dr_sandbox.andr.**manifest**.meta_data.resource(/pattern/)
meta_data.resource_num	Метаданные: ресурс.	dr_sandbox.andr.**manifest**.meta_data.resource_num
meta_data.value(regex)	Метаданные: значение.	dr_sandbox.andr.**manifest**.meta_data.value(/pattern/)
meta_data.value_num	Метаданные: значение.	dr_sandbox.andr.**manifest**.meta_data.value_num
package(regex)	Имя пакета приложения.	dr_sandbox.andr.**manifest**.package(/pattern/)
package_num	Имя пакета приложения.	dr_sandbox.andr.**manifest**.package_num
permissions(regex)	Список требуемых приложением разрешений.	dr_sandbox.andr.**manifest**.permissions(/pattern/)
permissions_num	Список требуемых приложением разрешений.	dr_sandbox.andr.**manifest**.permissions_num
receivers(regex)	Список широковещательных приемников.	dr_sandbox.andr.**manifest**.receivers(/pattern/)
receivers_num	Список широковещательных приемников.	dr_sandbox.andr.**manifest**.receivers_num
services(regex)	Список сервисов приложения.	dr_sandbox.andr.**manifest**.services(/pattern/)
services_num	Список сервисов приложения.	dr_sandbox.andr.**manifest**.services_num
strings_resources(regex)	Список всех строковых ресурсов.	dr_sandbox.andr.**manifest**.strings_resources(/pattern/)
strings_resources_num	Список всех строковых ресурсов.	dr_sandbox.andr.**manifest**.strings_resources_num
version_code(regex)	Код версии.	dr_sandbox.andr.**manifest**.version_code(/pattern/)

Функция	Результат	Примеры
`version_code_num`	Код версии.	`dr_sandbox.andr.manifest.version_code_num`
`version_name(regex)`	Имя версии.	`dr_sandbox.andr.manifest.version_name(/pattern/)`
`version_name_num`	Имя версии.	`dr_sandbox.andr.manifest.version_name_num`
`resources_digests(regex)`	Список SHA1-Digest файлов ресурсов APK.	`dr_sandbox.andr.resources_digests(/pattern/)`
`resources_digests_num`	Список SHA1-Digest файлов ресурсов APK.	`dr_sandbox.andr.resources_digests_num`
`sha1(regex)`	SHA1 семпла.	`dr_sandbox.andr.sha1(/pattern/)`
`sha1_num`	SHA1 семпла.	`dr_sandbox.andr.sha1_num`
`source_host(regex)`	Источник семпла.	`dr_sandbox.andr.source_host(/pattern/)`
`source_host_num`	Источник семпла.	`dr_sandbox.andr.source_host_num`

Описание функций для песочницы Windows (категория descr_tech)

Обеспечение автозапуска и распространения (категория autorun)

Функция	Результат	Тип события	Примеры
`change_system_executable_files(regex)`	Возвращает количество событий определенного типа.	Изменяет исполняемые системные файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.change_system_executable_files(/beep.sys/)`
`change_system_executable_files_num`	Возвращает количество событий определенного типа.	Изменяет исполняемые системные файлы.	`dr_sandbox.descr_tech.autorun.change_system_executable_files_num > 0`

Функция	Результат	Тип события	Примеры
`create_files_on_removable_media(regex)`	Возвращает количество событий определенного типа.	Создает файлы на съемном носителе. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.create_files_on_removable_media(/10thingscondoms.pdf/)`
`create_files_on_removable_media_num`	Возвращает количество событий определенного типа.	Создает файлы на съемном носителе.	`dr_sandbox.descr_tech.autorun.create_files_on_removable_media_num > 0`
`create_or_modify_files(regex)`	Возвращает количество событий определенного типа.	Создает или изменяет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.create_or_modify_files(/YogaGuide.job/)`
`create_or_modify_files_num`	Возвращает количество событий определенного типа.	Создает или изменяет файлы.	`dr_sandbox.descr_tech.autorun.create_or_modify_files_num == 1`
`create_services(regex)`	Возвращает количество событий определенного типа.	Создает сервисы. Учитываются только сервисы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.create_services(/rsdsys/)`
`create_services_num`	Возвращает количество событий определенного типа.	Создает сервисы.	`dr_sandbox.descr_tech.autorun.create_services_num > 0`
`infect_executables(regex)`	Возвращает количество событий определенного типа.	Заражает исполняемые файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.infect_executables(/eirmayxm/)`
`infect_executables_num`	Возвращает количество событий определенного типа.	Заражает исполняемые файлы.	`dr_sandbox.descr_tech.autorun.infect_executables_num > 0`

Функция	Результат	Тип события	Примеры
`modify_mbr`	Возвращает 1, если главная загрузочная запись (MBR) модифицирована, 0 — если нет.	Модифицирует главную загрузочную запись (MBR).	`dr_sandbox.descr_tech.autorun.modify_mbr`
`modify_registry(regex)`	Возвращает количество событий определенного типа.	Модифицирует ключи реестра. Учитываются только ключи, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.modify_registry(/C:\Users\user\AppData\Roaming\Sample.lnk/)`
`modify_registry_num`	Возвращает количество событий определенного типа.	Модифицирует ключи реестра.	`dr_sandbox.descr_tech.autorun.modify_registry_num >= 2`
`replace_system_executable_files(regex)`	Возвращает количество событий определенного типа.	Подменяет исполняемые системные файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.autorun.replace_system_executable_files(/ir50_qc.dll/)`
`replace_system_executable_files_num`	Возвращает количество событий определенного типа.	Подменяет исполняемые системные файлы.	`dr_sandbox.descr_tech.autorun.replace_system_executable_files_num > 0`

Изменения в файловой системе (категория filesystem)

Функция	Результат	Тип события	Примеры
`change_user_data_extensions`	Возвращает количество событий определенного типа.	Изменяет расширения файлов пользовательских данных (Trojan.Encoder).	`dr_sandbox.descr_tech.filesystem.change_user_data_extensions`
`create_files(regex)`	Возвращает количество событий определенного типа.	Создает файлы. Учитываются только файлы, удовлетворяющие	`dr_sandbox.descr_tech.filesystem.create_files(/nsArray.dll/)`

Функция	Результат	Тип события	Примеры
		регулярному выражению.	
create_files_num	Возвращает количество событий определенного типа.	Создает файлы.	dr_sandbox.descr_tech.filesystem.create_files_num >= 2
create_ransom_message_files	Возвращает количество событий определенного типа.	Создает файлы с требованием оплатить расшифровку файлов (Trojan.Encoder).	dr_sandbox.descr_tech.filesystem.create_ransom_message_files
modify_hosts	Возвращает 1, если файл HOSTS изменен, 0 — если нет.	Изменяет файл HOSTS.	dr_sandbox.descr_tech.filesystem.modify_hosts
modify_user_data_files	Возвращает количество событий определенного типа.	Изменяет множество файлов пользовательских данных (Trojan.Encoder).	dr_sandbox.descr_tech.filesystem.modify_user_data_files
move_files(regex)	Возвращает количество событий определенного типа.	Перемещает файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.filesystem.move_files(/%WINDIR%.*CONFIG\security.config.cch/)
move_files_num	Возвращает количество событий определенного типа.	Перемещает файлы.	dr_sandbox.descr_tech.filesystem.move_files_num >= 2
move_self(regex)	Возвращает количество событий определенного типа.	Самоперемещается.	dr_sandbox.descr_tech.filesystem.move_self(/CreativeAudio/)
move_self_num	Возвращает количество событий определенного типа.	Самоперемещается.	dr_sandbox.descr_tech.filesystem.move_self_num >= 2
move_system_files(regex)	Возвращает количество событий	Перемещает системные файлы. Учитываются только	dr_sandbox.descr_tech.filesystem.move_system_files(/ir50_qc.dll/)

Функция	Результат	Тип события	Примеры
	определенного типа.	файлы, удовлетворяющие регулярному выражению.	
`move_system_files_num`	Возвращает количество событий определенного типа.	Перемещает системные файлы.	`dr_sandbox.descr_tech.filesystem.move_system_files_num >= 2`
`remove_files(regex)`	Возвращает количество событий определенного типа.	Удаляет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.filesystem.remove_files(/^%TEMP%\7zS1.tmp\GOMPLAYERENSETUP.EXE$/)`
`remove_files_num`	Возвращает количество событий определенного типа.	Удаляет файлы.	`dr_sandbox.descr_tech.filesystem.remove_files_num >= 2`
`remove_self`	Возвращает количество событий определенного типа.	Самоудаляется.	`dr_sandbox.descr_tech.filesystem.remove_self`
`set_hidden(regex)`	Возвращает количество событий определенного типа.	Присваивает атрибут «скрытый» для файлов. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.filesystem.set_hidden(/^%TEMP%\~2.cmd$/)`
`set_hidden_num`	Возвращает количество событий определенного типа.	Присваивает атрибут «скрытый» для файлов.	`dr_sandbox.descr_tech.filesystem.set_hidden_num >= 2`
`substitute_executables(regex)`	Возвращает количество событий определенного типа.	Подменяет исполняемые файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.filesystem.substitute_executables(/pattern/)`
`substitute_executables_num`	Возвращает количество событий	Подменяет исполняемые	`dr_sandbox.descr_tech.filesystem.substitute_executables_`

Функция	Результат	Тип события	Примеры
	определенного типа.	файлы.	`num >= 2`
`substitute_files(regex)`	Возвращает количество событий определенного типа.	Подменяет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.filesystem.substitute_files(/pattern/)`
`substitute_files_num`	Возвращает количество событий определенного типа.	Подменяет файлы.	`dr_sandbox.descr_tech.filesystem.substitute_files_num >= 2`
`substitute_hosts`	Возвращает количество событий определенного типа.	Подменяет файл HOSTS.	`dr_sandbox.descr_tech.filesystem.substitute_hosts`

Вредоносные функции (категория malicious)

Функция	Результат	Тип события	Примеры
`add_antivirus_exclusion(regex)`	Возвращает количество событий определенного типа.	Чтобы затруднить выявление своего присутствия в системе, добавляет исключения антивируса с помощью ключей реестра. Учитываются только ключи, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.add_antivirus_exclusion(/pattern/)`
`add_antivirus_exclusion_num`	Возвращает количество событий определенного типа.	Чтобы затруднить выявление своего присутствия в системе, добавляет исключения антивируса с помощью ключей реестра.	`dr_sandbox.descr_tech.malicious.add_antivirus_exclusion_num`

Функция	Результат	Тип события	Примеры
block_cmd	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Интерпретатор командной строки (CMD).	dr_sandbox.descr_tech.malicious.block_cmd
block_regedit	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Редактор реестра (RegEdit).	dr_sandbox.descr_tech.malicious.block_regedit
block_system_file_checker	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Средство проверки системных файлов (SFC).	dr_sandbox.descr_tech.malicious.block_system_file_checker
block_system_restore	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Компонент восстановления системы (SR).	dr_sandbox.descr_tech.malicious.block_system_restore
block_taskmgr	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Диспетчер задач (Taskmgr).	dr_sandbox.descr_tech.malicious.block_taskmgr
block_user_account_control	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Средство контроля пользовательских	dr_sandbox.descr_tech.malicious.block_user_account_control

Функция	Результат	Тип события	Примеры
		учетных записей (UAC).	
block_windows_action_center	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Центр поддержки Windows (Action Center).	dr_sandbox.descr_tech.malicious.block_windows_action_center
block_windows_defender	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Системный антивирус (Защитник Windows).	dr_sandbox.descr_tech.malicious.block_windows_defender
block_windows_file_protection	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Систему защиты файлов операционной системы Windows (WFP).	dr_sandbox.descr_tech.malicious.block_windows_file_protection
block_windows_firewall	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Межсетевой экран (Брандмауэр Windows).	dr_sandbox.descr_tech.malicious.block_windows_firewall
block_windows_security_center	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует Центр обеспечения безопасности (Security Center).	dr_sandbox.descr_tech.malicious.block_windows_security_center

Функция	Результат	Тип события	Примеры
`block_windows_updates`	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует запуск следующей системной утилиты: Обновления системы (Windows Update).	`dr_sandbox.descr_tech.malicious.block_windows_updates`
`bruteforce_os_accounts`	Возвращает 1, если событие произошло, 0 — если нет.	Перебирает пароли аккаунтов ОС.	`dr_sandbox.descr_tech.malicious.bruteforce_os_accounts`
`create_and_exec(regex)`	Возвращает количество событий определенного типа.	Создает и запускает на исполнение. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.create_and_exec(/Total Commander/)`
`create_and_exec_num`	Возвращает количество событий определенного типа.	Создает и запускает на исполнение.	`dr_sandbox.descr_tech.malicious.create_and_exec_num > 0`
`create_onion_service`	Возвращает количество событий определенного типа.	Создает onion-сервис.	`dr_sandbox.descr_tech.malicious.create_onion_service`
`delete_volume_shadow_copies`	Возвращает количество событий определенного типа.	Для затруднения выявления своего присутствия в системе удаляет теневые копии разделов.	`dr_sandbox.descr_tech.malicious.delete_volume_shadow_copies`
`detect_virtual_machine(regex)`	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения виртуальных машин. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.detect_virtual_machine(/pattern/)`

Функция	Результат	Тип события	Примеры
detect_virtual_machine_num	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения виртуальных машин.	dr_sandbox.descr_tech.malicious.detect_virtual_machine_num
disable_amsi	Возвращает количество событий определенного типа.	Отключает AMSI.	dr_sandbox.descr_tech.malicious.disable_amsi
downloads_and_executes(regex)	Возвращает количество событий определенного типа.	Загружает и запускает на исполнение. Учитываются только объекты, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.downloads_and_executes(/pattern/)
downloads_and_executes_num	Возвращает количество событий определенного типа.	Загружает и запускает на исполнение.	dr_sandbox.descr_tech.malicious.downloads_and_executes_num
downloads_and_executes_files	Возвращает количество событий определенного типа.	Загружает и запускает на исполнение файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.downloads_and_executes_files
download_file(regex)	Возвращает количество событий определенного типа.	Загружает файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.download_file(/pattern/)
download_file_num	Возвращает количество событий определенного типа.	Загружает файлы.	dr_sandbox.descr_tech.malicious.download_file_num
download_files	Возвращает 1, если событие произошло, 0 — если нет.	Загружает файлы.	dr_sandbox.descr_tech.malicious.download_files

Функция	Результат	Тип события	Примеры
exec(regex)	Возвращает количество событий определенного типа.	Запускает на исполнение. Учитываются только объекты, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.exec(/netsh.exe/)
exec_num	Возвращает количество событий определенного типа.	Запускает на исполнение.	dr_sandbox.descr_tech.malicious.exec_num > 0
exec_wmi(regex)	Возвращает количество событий определенного типа.	Выполняет операции WMI. Учитываются только операции, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.exec_wmi(/pattern/)
exec_wmi_num	Возвращает количество событий определенного типа.	Выполняет операции WMI.	dr_sandbox.descr_tech.malicious.exec_wmi_num
exploit_create_and_exec(regex)	Возвращает количество событий определенного типа.	Создает и запускает на исполнение (эксплойт). Учитываются только объекты, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.exploit_create_and_exec(/pattern/)
exploit_create_and_exec_num	Возвращает количество событий определенного типа.	Создает и запускает на исполнение (эксплойт).	dr_sandbox.descr_tech.malicious.exploit_create_and_exec_num
exploit_create_and_load_library(regex)	Возвращает количество событий определенного типа.	Создает и загружает библиотеки (эксплойт). Учитываются только библиотеки, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.exploit_create_and_load_library(/pattern/)
exploit_create_and_load_li	Возвращает количество событий	Создает и загружает библиотеки	dr_sandbox.descr_tech.malicious.exploit_create_and_load_

Функция	Результат	Тип события	Примеры
brary_num	определенного типа.	(эксплойт).	library_num
exploit_exec(regex)	Возвращает количество событий определенного типа.	Запускает на исполнение (эксплойт). Учитываются только объекты, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.exploit_exec(/pattern/)
exploit_exec_num	Возвращает количество событий определенного типа.	Запускает на исполнение (эксплойт).	dr_sandbox.descr_tech.malicious.exploit_exec_num
force_autorun_for_removable_media	Возвращает 1, если событие произошло, 0 — если нет.	Принудительно разрешает автозапуск со съемных носителей.	dr_sandbox.descr_tech.malicious.force_autorun_for_removable_media
hide_from_view_file_extensions	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует отображение расширений файлов.	dr_sandbox.descr_tech.malicious.hide_from_view_file_extensions
hide_from_view_hidden_files	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе блокирует отображение скрытых файлов.	dr_sandbox.descr_tech.malicious.hide_from_view_hidden_files
hide_processes(regex)	Возвращает количество событий определенного типа.	Скрывает процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.hide_processes(/cscript.exe/)
hide_processes_num	Возвращает количество событий определенного типа.	Скрывает процессы.	dr_sandbox.descr_tech.malicious.hide_processes_num > 0

Функция	Результат	Тип события	Примеры
`hide_taskbar_notifications`	Возвращает 1, если событие произошло, 0 — если нет.	Для затруднения выявления своего присутствия в системе отключает уведомления панели задач.	`dr_sandbox.descr_tech.malicious.hide_taskbar_notifications`
`hook_in_browser(regex)`	Возвращает количество событий определенного типа.	Перехватывает функции в браузерах. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.hook_in_browser(/pattern/)`
`hook_in_browser_num`	Возвращает количество событий определенного типа.	Перехватывает функции в браузерах.	`dr_sandbox.descr_tech.malicious.hook_in_browser_num`
`hook_keyboard_all_processes(regex)`	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата сообщений о нажатии клавиш клавиатуры: Библиотека-обработчик для всех процессов: (?LibraryPath).	`dr_sandbox.descr_tech.malicious.hook_keyboard_all_processes(/OQKWHP\BJX.01/)`
`hook_keyboard_all_processes_num`	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата сообщений о нажатии клавиш клавиатуры.	`dr_sandbox.descr_tech.malicious.hook_keyboard_all_processes_num > 0`
`hook_keyboard_concrete_processes(regex)`	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата сообщений о нажатии клавиш клавиатуры: Библиотека-обработчик для процесса '(?	`dr_sandbox.descr_tech.malicious.hook_keyboard_concrete_processes(/IMDCSC.exe/)`

Функция	Результат	Тип события	Примеры
		HookedProcess.Name)': (?LibraryPath).	
hook_keyboard_concrete_processes_num	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата сообщений о нажатии клавиш клавиатуры.	dr_sandbox.descr_tech.malicious.hook_keyboard_concrete_processes_num > 0
hook_keyboard_on_window_messages(regex)	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата оконных сообщений. Учитываются только объекты, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.hook_keyboard_on_window_messages(/pattern/)
hook_keyboard_on_window_messages_num	Возвращает количество событий определенного типа.	Устанавливает процедуры перехвата оконных сообщений.	dr_sandbox.descr_tech.malicious.hook_keyboard_on_window_messages_num
inject_to_a_lot_of_user_processes	Возвращает 1, если событие произошло, 0 — если нет.	Внедряет код в большое количество пользовательских процессов.	dr_sandbox.descr_tech.malicious.inject_to_a_lot_of_user_processes
inject_to_system_proc(regex)	Возвращает количество событий определенного типа.	Внедряет код в системные процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.inject_to_system_proc(/RegAsm.exe/)
inject_to_system_proc_num	Возвращает количество событий определенного типа.	Внедряет код в системные процессы.	dr_sandbox.descr_tech.malicious.inject_to_system_proc_num > 0
inject_to_user_proc(regex)	Возвращает количество событий определенного типа.	Внедряет код в пользовательские процессы. Учитываются только процессы, удовлетворяющие	dr_sandbox.descr_tech.malicious.inject_to_user_proc(/^iexplore.exe$/)

Функция	Результат	Тип события	Примеры
		регулярному выражению.	
inject_to_user_proc_num	Возвращает количество событий определенного типа.	Внедряет код в пользовательские процессы.	dr_sandbox.descr_tech.malicious.inject_to_user_proc_num > 0
modify_explorer_settings(regex)	Возвращает количество событий определенного типа.	Изменяет настройки проводника Windows (Windows Explorer). Учитываются только настройки, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.modify_explorer_settings(/'NoFolderOptions' = '00000001'/)
modify_explorer_settings_num	Возвращает количество событий определенного типа.	Изменяет настройки проводника Windows (Windows Explorer).	dr_sandbox.descr_tech.malicious.modify_explorer_settings_num > 0
modify_ie_settings(regex)	Возвращает количество событий определенного типа.	Изменяет настройки браузера Windows Internet Explorer. Учитываются только настройки, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.modify_ie_settings(/Zones\1] '1206' = '00000000'/)
modify_ie_settings_num	Возвращает количество событий определенного типа.	Изменяет настройки браузера Windows Internet Explorer.	dr_sandbox.descr_tech.malicious.modify_ie_settings_num > 0
modify_registry_to_bypass_firewall(regex)	Возвращает количество событий определенного типа.	Для обхода брандмауэра удаляет или модифицирует определенные ключи реестра. Учитываются только ключи, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.modify_registry_to_bypass_firewall(/Enabled:taskmg.exe/)
modify_registry_to_bypass_	Возвращает количество событий	Для обхода брандмауэра	dr_sandbox.descr_tech.malicious.modify_registry_to_bypas

Функция	Результат	Тип события	Примеры
`firewall_num`	определенного типа.	удаляет или модифицирует определенные ключи реестра.	`s_firewall_num > 0`
`modify_system_dns(regex)`	Возвращает количество событий определенного типа.	Для затруднения выявления своего присутствия в системе изменяет DNS-серверы. Учитываются только серверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.modify_system_dns(/pattern/)`
`modify_system_dns_num`	Возвращает количество событий определенного типа.	Для затруднения выявления своего присутствия в системе изменяет DNS-серверы.	`dr_sandbox.descr_tech.malicious.modify_system_dns_num`
`modify_system_settings(regex)`	Возвращает количество событий определенного типа.	Для затруднения выявления своего присутствия в системе изменяет системные настройки. Учитываются только настройки, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.modify_system_settings(/pattern/)`
`modify_system_settings_num`	Возвращает количество событий определенного типа.	Для затруднения выявления своего присутствия в системе изменяет системные настройки.	`dr_sandbox.descr_tech.malicious.modify_system_settings_num`
`read_third_party_passwords(regex)`	Возвращает количество событий определенного типа.	Читает файлы, отвечающие за хранение паролей сторонними программами. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.read_third_party_passwords(/pattern/)`

Функция	Результат	Тип события	Примеры
`read_third_party_passwords_num`	Возвращает количество событий определенного типа.	Читает файлы, отвечающие за хранение паролей сторонними программами.	`dr_sandbox.descr_tech.malicious.read_third_party_passwords_num`
`register_bho(regex)`	Возвращает количество событий определенного типа.	Регистрирует BHO. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.register_bho(/pattern/)`
`register_com_server(regex)`	Возвращает количество событий определенного типа.	Регистрирует COM-сервер. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.register_com_server(/pattern/)`
`register_com_server_num`	Возвращает количество событий определенного типа.	Регистрирует COM-сервер.	`dr_sandbox.descr_tech.malicious.register_com_server_num`
`register_filesystem_filter(regex)`	Возвращает количество событий определенного типа.	Регистрирует фильтр файловой системы. Учитываются только объекты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.register_filesystem_filter(/pattern/)`
`restore_ssdt_hooks`	Возвращает 1, если событие произошло, 0 — если нет.	Устраняет перехваты функций в SSDT (System Service Descriptor Table).	`dr_sandbox.descr_tech.malicious.restore_ssdt_hooks`
`search_password_in_registry(regex)`	Возвращает количество событий определенного типа.	Ищет ветки реестра, отвечающие за хранение паролей сторонними программами. Учитываются только ветки, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.search_password_in_registry(/MessengerService/)`

Функция	Результат	Тип события	Примеры
search_password_in_registry_num	Возвращает количество событий определенного типа.	Ищет ветки реестра, отвечающие за хранение паролей сторонними программами.	dr_sandbox.descr_tech.malicious.search_password_in_registry_num > 0
search_wnd_for_analyzing_soft(regex)	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения утилит для анализа. Учитываются только окна, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.search_wnd_for_analyzing_soft(/PEiD/)
search_wnd_for_analyzing_soft_num	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения утилит для анализа.	dr_sandbox.descr_tech.malicious.search_wnd_for_analyzing_soft_num > 0
search_wnd_for_programs_and_games(regex)	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения различных программ и игр. Учитываются только окна, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.search_wnd_for_programs_and_games(/The Wireshark Network Analyzer/)
search_wnd_for_programs_and_games_num	Возвращает количество событий определенного типа.	Ищет окна с целью обнаружения различных программ и игр.	dr_sandbox.descr_tech.malicious.search_wnd_for_programs_and_games_num > 0
search_wnd_to_bypass_av(regex)	Возвращает количество событий определенного типа.	Ищет окна с целью обхода различных антивирусов. Учитываются только окна, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.search_wnd_to_bypass_av(/AVP.AlertDialog/)
search_wnd_to_bypass_av_num	Возвращает количество событий определенного типа.	Ищет окна с целью обхода различных антивирусов.	dr_sandbox.descr_tech.malicious.search_wnd_to_bypass_av_num > 0
search_wnd_to	Возвращает количество событий	Ищет окна с целью обхода системы	dr_sandbox.descr_tech.malicious.search_wnd_to_bypass_wfp

Функция	Результат	Тип события	Примеры
_bypass_wfp(regex)	определенного типа.	защиты файлов Windows (WFP). Учитываются только окна, удовлетворяющие регулярному выражению.	(/Windows File Protection/)
search_wnd_to_bypass_wfp_num	Возвращает количество событий определенного типа.	Ищет окна с целью обхода системы защиты файлов Windows (WFP).	dr_sandbox.descr_tech.malicious.search_wnd_to_bypass_wfp_num > 0
set_concrete_ssdt_hooks(regex)	Возвращает количество событий определенного типа.	Перехватывает функции в SSDT (System Service Descriptor Table). Учитываются только функции, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech.malicious.set_concrete_ssdt_hooks(/pattern/)
set_concrete_ssdt_hooks_num	Возвращает количество событий определенного типа.	Перехватывает функции в SSDT (System Service Descriptor Table).	dr_sandbox.descr_tech.malicious.set_concrete_ssdt_hooks_num
set_homepage_for_ie	Возвращает 1, если событие произошло, 0 — если нет.	Без разрешения пользователя устанавливает новую стартовую страницу для Windows Internet Explorer.	dr_sandbox.descr_tech.malicious.set_homepage_for_ie
set_ssdt_hooks	Возвращает количество событий определенного типа.	Перехватывает функции в SSDT (System Service Descriptor Table).	dr_sandbox.descr_tech.malicious.set_ssdt_hooks
try_to_terminate_a_lot_of_user_processes	Возвращает 1, если событие произошло, 0 — если нет.	Завершает или пытается завершить большое количество пользовательских процессов.	dr_sandbox.descr_tech.malicious.try_to_terminate_a_lot_of_user_processes
try_to_terminate_system_pr	Возвращает количество событий	Завершает или пытается завершить системные	dr_sandbox.descr_tech.malicious.try_to_terminate_system_processes(/ctfmon.exe/)

Функция	Результат	Тип события	Примеры
`ocesses(regex)`	определенного типа.	процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	
`try_to_terminate_system_processes_num`	Возвращает количество событий определенного типа.	Завершает или пытается завершить системные процессы.	`dr_sandbox.descr_tech.malicious.try_to_terminate_system_processes_num > 0`
`try_to_terminate_user_processes(regex)`	Возвращает количество событий определенного типа.	Завершает или пытается завершить пользовательские процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.malicious.try_to_terminate_user_processes(/^AVSYNMGR.EXE$/)`
`try_to_terminate_user_processes_num`	Возвращает количество событий определенного типа.	Завершает или пытается завершить пользовательские процессы.	`dr_sandbox.descr_tech.malicious.try_to_terminate_user_processes_num > 0`

Другое (категория miscellaneous)

Функция	Результат	Тип события	Примеры
`add_root_certificate`	Возвращает 1, если сертификат добавлен, 0 — если нет.	Добавляет корневой сертификат.	`dr_sandbox.descr_tech.miscellaneous.add_root_certificate`
`create_and_exec`	Возвращает 1, если событие произошло, 0 — если нет.	Создает и запускает на исполнение (со скрытым окном).	`dr_sandbox.descr_tech.miscellaneous.create_and_exec`
`disable_certificate`	Возвращает 1, если событие произошло, 0 — если нет.	Отключает сертификат.	`dr_sandbox.descr_tech.miscellaneous.disable_certificate`
`exec(regex)`	Возвращает количество событий	Запускает на исполнение.	`dr_sandbox.descr_tech.miscellaneous.exec(/pattern/)`

Функция	Результат	Тип события	Примеры
	определенного типа.	Учитываются только процессы, удовлетворяющие регулярному выражению.	
`load_driver(regex)`	Возвращает количество событий определенного типа.	Загружает драйверы. Учитываются только драйверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.miscellaneous.load_driver(/pattern/)`
`load_driver_num`	Возвращает количество событий определенного типа.	Загружает драйверы.	`dr_sandbox.descr_tech.miscellaneous.load_driver_num`
`modify_auto_config_url(regex)`	Возвращает количество событий определенного типа.	Изменяет значение AutoConfigURL на новое. Учитываются только новые значения, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.miscellaneous.modify_auto_config_url(/pattern/)`
`search_wnd(regex)`	Возвращает количество событий определенного типа.	Ищет окна. Учитываются только окна, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech.miscellaneous.search_wnd(/MS_WebcheckMonitor/)`
`search_wnd_num`	Возвращает количество событий определенного типа.	Ищет окна.	`dr_sandbox.descr_tech.miscellaneous.search_wnd_num == 3`
`shut_down_windows`	Возвращает 1, если событие произошло, 0 — если нет.	Пытается завершить работу операционной системы Windows.	`dr_sandbox.descr_tech.miscellaneous.shut_down_windows`
`use_ntfs_data_streams`	Возвращает 1, если событие произошло, 0 — если нет.	Использует альтернативные потоки данных NTFS.	`dr_sandbox.descr_tech.miscellaneous.use_ntfs_data_streams`

Сетевая активность (категория network)

Функция	Результат	Тип события	Примеры
`connect_to(regex)`	Возвращает количество событий определенного типа.	Подключается к перечисленному в регулярном выражении.	`dr_sandbox.descr_tech.network.connect_to(/www.xfo.cn/)`
`connect_to_num`	Возвращает количество событий определенного типа.	Подключается.	`dr_sandbox.descr_tech.network.connect_to_num >= 2`
`tcp(regex)`	Возвращает количество событий определенного типа.	Запросы через TCP.	`dr_sandbox.descr_tech.network.tcp(/pattern/)`
`tcp_num`	Возвращает количество событий определенного типа.	Запросы через TCP.	`dr_sandbox.descr_tech.network.tcp_num`
`tcp_http_get(regex)`	Возвращает количество событий определенного типа.	Запросы HTTP GET через TCP.	`dr_sandbox.descr_tech.network.tcp_http_get(/addurl.html$/)`
`tcp_http_get_num`	Возвращает количество событий определенного типа.	Запросы HTTP GET через TCP.	`dr_sandbox.descr_tech.network.tcp_http_get_num >= 2`
`tcp_http_post(regex)`	Возвращает количество событий определенного типа.	Запросы HTTP POST через TCP.	`dr_sandbox.descr_tech.network.tcp_http_post(/addurl.html$/)`
`tcp_http_post_num`	Возвращает количество событий определенного типа.	Запросы HTTP POST через TCP.	`dr_sandbox.descr_tech.network.tcp_http_post_num >= 2`
`tcp_http_unk(regex)`	Возвращает количество событий определенного типа.	Запросы HTTP неизвестного формата.	`dr_sandbox.descr_tech.network.tcp_http_unk(/pattern/)`
`tcp_http_unk_num`	Возвращает количество событий	Запросы HTTP неизвестного формата.	`dr_sandbox.descr_tech.network.tcp_http_unk_num`

Функция	Результат	Тип события	Примеры
	определенного типа.		
`udp(regex)`	Возвращает количество событий определенного типа.	Запросы через UDP.	`dr_sandbox.descr_tech.network.udp(/disk57/)`
`udp_num`	Возвращает количество событий определенного типа.	Запросы через UDP.	`dr_sandbox.descr_tech.network.udp_num >= 2`

Описание функций для песочницы Linux (категория descr_tech_lbcl)

Обеспечение автозапуска и распространения (категория autorun)

Функция	Результат	Тип события	Примеры
`create_or_modify_files(regex)`	Возвращает количество событий определенного типа.	Создает или изменяет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.autorun.create_or_modify_files(/pattern/)`
`create_or_modify_files_num`	Возвращает количество событий определенного типа.	Создает или изменяет файлы.	`dr_sandbox.descr_tech_lbcl.autorun.create_or_modify_files_num`
`create_or_modify_symlinks(regex)`	Возвращает количество событий определенного типа.	Создает или изменяет символические ссылки. Учитываются только ссылки, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.autorun.create_or_modify_symlinks(/pattern/)`
`create_or_modify_symlinks_num`	Возвращает количество событий	Создает или изменяет	`dr_sandbox.descr_tech_lbcl.autorun.create_or_modify_symlinks_num`

Функция	Результат	Тип события	Примеры
	определенного типа.	символические ссылки.	

Изменения в файловой системе (категория filesystem)

Функция	Результат	Тип события	Примеры
`change_time_of_file(regex)`	Возвращает количество событий определенного типа.	Изменяет время создания, доступа, модификации файлов. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.change_time_of_file(/pattern/)`
`change_time_of_file_num`	Возвращает количество событий определенного типа.	Изменяет время создания, доступа, модификации файлов.	`dr_sandbox.descr_tech_lbcl.filesystem.change_time_of_file_num`
`create_dir(regex)`	Возвращает количество событий определенного типа.	Создает каталоги. Учитываются только каталоги, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.create_dir(/pattern/)`
`create_dir_num`	Возвращает количество событий определенного типа.	Создает каталоги.	`dr_sandbox.descr_tech_lbcl.filesystem.create_dir_num`
`create_or_modify_file(regex)`	Возвращает количество событий определенного типа.	Создает или изменяет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.create_or_modify_file(/pattern/)`
`create_or_modify_file_num`	Возвращает количество событий определенного типа.	Создает или изменяет файлы.	`dr_sandbox.descr_tech_lbcl.filesystem.create_or_modify_file_num`

Функция	Результат	Тип события	Примеры
`create_symlink(regex)`	Возвращает количество событий определенного типа.	Создает символические ссылки.	`dr_sandbox.descr_tech_lbcl.filesystem.create_symlink(/pattern/)`
`create_symlink_num`	Возвращает количество событий определенного типа.	Учитываются только ссылки, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.create_symlink_num`
`lock_file(regex)`	Возвращает количество событий определенного типа.	Блокирует файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.lock_file(/pattern/)`
`lock_file_num`	Возвращает количество событий определенного типа.	Блокирует файлы.	`dr_sandbox.descr_tech_lbcl.filesystem.lock_file_num`
`modify_file_access_rights(regex)`	Возвращает количество событий определенного типа.	Изменяет права доступа к файлам.	`dr_sandbox.descr_tech_lbcl.filesystem.modify_file_access_rights(/pattern/)`
`modify_file_access_rights_num`	Возвращает количество событий определенного типа.	Изменяет права доступа к файлам.	`dr_sandbox.descr_tech_lbcl.filesystem.modify_file_access_rights_num`
`modify_file_owner(regex)`	Возвращает количество событий определенного типа.	Изменяет владельца файлов.	`dr_sandbox.descr_tech_lbcl.filesystem.modify_file_owner(/pattern/)`
`modify_file_owner_num`	Возвращает количество событий определенного типа.	Изменяет владельца файлов.	`dr_sandbox.descr_tech_lbcl.filesystem.modify_file_owner_num`
`mount_file_system(regex)`	Возвращает количество событий определенного типа.	Монтирует файловые системы. Учитываются только системы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.mount_file_system(/pattern/)`

Функция	Результат	Тип события	Примеры
`mount_file_system_num`	Возвращает количество событий определенного типа.	Монтирует файловые системы.	`dr_sandbox.descr_tech_lbcl.filesystem.mount_file_system_num`
`remove_dir(regex)`	Возвращает количество событий определенного типа.	Удаляет каталоги. Учитываются только каталоги, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.remove_dir(/pattern/)`
`remove_dir_num`	Возвращает количество событий определенного типа.	Удаляет каталоги.	`dr_sandbox.descr_tech_lbcl.filesystem.remove_dir_num`
`remove_file(regex)`	Возвращает количество событий определенного типа.	Удаляет файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.remove_file(/pattern/)`
`remove_file_num`	Возвращает количество событий определенного типа.	Удаляет файлы.	`dr_sandbox.descr_tech_lbcl.filesystem.remove_file_num`
`unmount_file_system(regex)`	Возвращает количество событий определенного типа.	Демонтирует файловые системы. Учитываются только системы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.filesystem.unmount_file_system(/pattern/)`
`unmount_file_system_num`	Возвращает количество событий определенного типа.	Демонтирует файловые системы.	`dr_sandbox.descr_tech_lbcl.filesystem.unmount_file_system_num`

Вредоносные функции (категория malicious)

Функция	Результат	Тип события	Примеры
`attempt_kill_`	Возвращает количество событий	Пытается завершить системные	`dr_sandbox.descr_tech_lbcl.malicious.attempt_kill_system`

Функция	Результат	Тип события	Примеры
system_proc(regex)	определенного типа.	процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	_proc(/pattern/)
attempt_kill_system_proc_num	Возвращает количество событий определенного типа.	Пытается завершить системные процессы.	dr_sandbox.descr_tech_lbcl.malicious.attempt_kill_system_proc_num
attept_kill_analyzers(regex)	Возвращает количество событий определенного типа.	Пытается завершить приложения-анализаторы. Учитываются только приложения, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.attept_kill_analyzers(/pattern/)
attept_kill_analyzers_num	Возвращает количество событий определенного типа.	Пытается завершить приложения-анализаторы.	dr_sandbox.descr_tech_lbcl.malicious.attept_kill_analyzers_num
attept_kill_proc(regex)	Возвращает количество событий определенного типа.	Пытается завершить процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.attept_kill_proc(/pattern/)
attept_kill_proc_num	Возвращает количество событий определенного типа.	Пытается завершить процессы.	dr_sandbox.descr_tech_lbcl.malicious.attept_kill_proc_num
compile_program_from_source_codes(regex)	Возвращает количество событий определенного типа.	Компилирует исходный код.	dr_sandbox.descr_tech_lbcl.malicious.compile_program_from_source_codes(/pattern/)
compile_program_from_source_codes_num	Возвращает количество событий определенного типа.	Компилирует исходный код.	dr_sandbox.descr_tech_lbcl.malicious.compile_program_from_source_codes_num

Функция	Результат	Тип события	Примеры
`gain_root_privileges`	Возвращает количество событий определенного типа.	Получает права суперпользователя (root).	`dr_sandbox.descr_tech_lbcl.malicious.gain_root_privileges`
`get_access_to_ssh_keys`	Возвращает количество событий определенного типа.	Получает доступ к ключам SSH.	`dr_sandbox.descr_tech_lbcl.malicious.get_access_to_ssh_keys`
`inject_to_proc(regex)`	Возвращает количество событий определенного типа.	Встраивается в процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.inject_to_proc(/pattern/)`
`inject_to_proc_num`	Возвращает количество событий определенного типа.	Встраивается в процессы.	`dr_sandbox.descr_tech_lbcl.malicious.inject_to_proc_num`
`kill_analyzers(regex)`	Возвращает количество событий определенного типа.	Завершает приложения-анализаторы. Учитываются только приложения, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.kill_analyzers(/pattern/)`
`kill_analyzers_num`	Возвращает количество событий определенного типа.	Завершает приложения-анализаторы.	`dr_sandbox.descr_tech_lbcl.malicious.kill_analyzers_num`
`kill_proc(regex)`	Возвращает количество событий определенного типа.	Завершает процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.kill_proc(/pattern/)`
`kill_proc_num`	Возвращает количество событий определенного типа.	Завершает процессы.	`dr_sandbox.descr_tech_lbcl.malicious.kill_proc_num`

Функция	Результат	Тип события	Примеры
`kill_system_proc(regex)`	Возвращает количество событий определенного типа.	Завершает системные процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.kill_system_proc(/pattern/)`
`kill_system_proc_num`	Возвращает количество событий определенного типа.	Завершает системные процессы.	`dr_sandbox.descr_tech_lbcl.malicious.kill_system_proc_num`
`launch_itself_as_daemon`	Возвращает количество событий определенного типа.	Запускает себя как управляющую программу.	`dr_sandbox.descr_tech_lbcl.malicious.launch_itself_as_daemon`
`launch_processes(regex)`	Возвращает количество событий определенного типа.	Запускает процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.launch_processes(/pattern/)`
`launch_processes_num`	Возвращает количество событий определенного типа.	Запускает процессы.	`dr_sandbox.descr_tech_lbcl.malicious.launch_processes_num`
`manage_services(regex)`	Возвращает количество событий определенного типа.	Управляет службами. Учитываются только службы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.manage_services(/pattern/)`
`manage_services_num`	Возвращает количество событий определенного типа.	Управляет службами.	`dr_sandbox.descr_tech_lbcl.malicious.manage_services_num`
`modify_firewall_settings(regex)`	Возвращает количество событий определенного типа.	Изменяет настройки брандмауэра. Учитываются только настройки, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.malicious.modify_firewall_settings(/pattern/)`

Функция	Результат	Тип события	Примеры
modify_firewall_settings_num	Возвращает количество событий определенного типа.	Изменяет настройки брандмауэра.	dr_sandbox.descr_tech_lbcl.malicious.modify_firewall_settings_num
modify_router_settings(regex)	Возвращает количество событий определенного типа.	Изменяет настройки маршрутизатора. Учитываются только настройки, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.modify_router_settings(/pattern/)
modify_router_settings_num	Возвращает количество событий определенного типа.	Изменяет настройки маршрутизатора.	dr_sandbox.descr_tech_lbcl.malicious.modify_router_settings_num
operate_kernel_modules(regex)	Возвращает количество событий определенного типа.	Использует модули ядра.	dr_sandbox.descr_tech_lbcl.malicious.operate_kernel_modules(/pattern/)
operate_kernel_modules_num	Возвращает количество событий определенного типа.	Использует модули ядра.	dr_sandbox.descr_tech_lbcl.malicious.operate_kernel_modules_num
perform_process_tracing(regex)	Возвращает количество событий определенного типа.	Отслеживает процессы. Учитываются только процессы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.perform_process_tracing(/pattern/)
perform_process_tracing_num	Возвращает количество событий определенного типа.	Отслеживает процессы.	dr_sandbox.descr_tech_lbcl.malicious.perform_process_tracing_num
remove_self	Возвращает количество событий определенного типа.	Самоудаляется.	dr_sandbox.descr_tech_lbcl.malicious.remove_self
remove_system_files(regex)	Возвращает количество событий определенного типа.	Удаляет системные файлы. Учитываются только файлы, удовлетворяющие	dr_sandbox.descr_tech_lbcl.malicious.remove_system_files(/pattern/)

Функция	Результат	Тип события	Примеры
		регулярному выражению.	
remove_system_files_num	Возвращает количество событий определенного типа.	Удаляет системные файлы.	dr_sandbox.descr_tech_lbcl.malicious.remove_system_files_num
replace_system_files(regex)	Возвращает количество событий определенного типа.	Заменяет системные файлы. Учитываются только файлы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.replace_system_files(/pattern/)
replace_system_files_num	Возвращает количество событий определенного типа.	Заменяет системные файлы.	dr_sandbox.descr_tech_lbcl.malicious.replace_system_files_num
stops_system_services(regex)	Возвращает количество событий определенного типа.	Останавливает системные службы. Учитываются только службы, удовлетворяющие регулярному выражению.	dr_sandbox.descr_tech_lbcl.malicious.stops_system_services(/pattern/)
stops_system_services_num	Возвращает количество событий определенного типа.	Останавливает системные службы.	dr_sandbox.descr_tech_lbcl.malicious.stops_system_services_num
substitute_application_name_for(regex)	Возвращает количество событий определенного типа.	Подменяет имя приложения.	dr_sandbox.descr_tech_lbcl.malicious.substitute_application_name_for(/pattern/)
substitute_application_name_for_num	Возвращает количество событий определенного типа.	Подменяет имя приложения.	dr_sandbox.descr_tech_lbcl.malicious.substitute_application_name_for_num

Сетевая активность (категория network)

Функция	Результат	Тип события	Примеры
`attack_bruteforce_via_ssh`	Возвращает количество событий определенного типа.	Проводит атаку перебором по SSH.	`dr_sandbox.descr_tech_lbcl.network.attack_bruteforce_via_ssh`
`attack_bruteforce_via_telnet`	Возвращает количество событий определенного типа.	Проводит атаку перебором по TELNET.	`dr_sandbox.descr_tech_lbcl.network.attack_bruteforce_via_telnet`
`attack_bruteforce_via_unk_protocol`	Возвращает количество событий определенного типа.	Проводит атаку перебором по неизвестному протоколу.	`dr_sandbox.descr_tech_lbcl.network.attack_bruteforce_via_unk_protocol`
`connect_to(regex)`	Возвращает количество событий определенного типа.	Подключается к серверам. Учитываются только серверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.network.connect_to(/pattern/)`
`connect_to_num`	Возвращает количество событий определенного типа.	Подключается к серверам.	`dr_sandbox.descr_tech_lbcl.network.connect_to_num`
`connect_to_irc(regex)`	Возвращает количество событий определенного типа.	Подключается к серверам по протоколу IRC. Учитываются только серверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.network.connect_to_irc(/pattern/)`
`dns_ask(regex)`	Возвращает количество событий определенного типа.	DNS-запросы.	`dr_sandbox.descr_tech_lbcl.network.dns_ask(/pattern/)`
`dns_ask_num`	Возвращает количество событий определенного типа.	DNS-запросы.	`dr_sandbox.descr_tech_lbcl.network.dns_ask_num`

Функция	Результат	Тип события	Примеры
`http_get(regex)`	Возвращает количество событий определенного типа.	Запросы HTTP GET.	`dr_sandbox.descr_tech_lbcl.network.http_get(/pattern/)`
`http_get_num`	Возвращает количество событий определенного типа.	Запросы HTTP GET.	`dr_sandbox.descr_tech_lbcl.network.http_get_num`
`http_other(regex)`	Возвращает количество событий определенного типа.	Другие запросы HTTP.	`dr_sandbox.descr_tech_lbcl.network.http_other(/pattern/)`
`http_other_num`	Возвращает количество событий определенного типа.	Другие запросы HTTP.	`dr_sandbox.descr_tech_lbcl.network.http_other_num`
`http_post(regex)`	Возвращает количество событий определенного типа.	Запросы HTTP POST.	`dr_sandbox.descr_tech_lbcl.network.http_post(/pattern/)`
`http_post_num`	Возвращает количество событий определенного типа.	Запросы HTTP POST.	`dr_sandbox.descr_tech_lbcl.network.http_post_num`
`listening_port(regex)`	Возвращает количество событий определенного типа.	Ждет входящие подключения на портах. Учитываются только порты, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.network.listening_port(/pattern/)`
`listening_port_num`	Возвращает количество событий определенного типа.	Ждет входящие подключения на портах.	`dr_sandbox.descr_tech_lbcl.network.listening_port_num`
`receive_data_from_server(regex)`	Возвращает количество событий определенного типа.	Получает данные с серверов. Учитываются только серверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.network.receive_data_from_server(/pattern/)`

Функция	Результат	Тип события	Примеры
`receive_data_from_server_num`	Возвращает количество событий определенного типа.	Получает данные с серверов.	`dr_sandbox.descr_tech_lbcl.network.receive_data_from_server_num`
`send_data_to_server(regex)`	Возвращает количество событий определенного типа.	Отправляет данные на серверы. Учитываются только серверы, удовлетворяющие регулярному выражению.	`dr_sandbox.descr_tech_lbcl.network.send_data_to_server(/pattern/)`
`send_data_to_server_num`	Возвращает количество событий определенного типа.	Отправляет данные на серверы.	`dr_sandbox.descr_tech_lbcl.network.send_data_to_server_num`

Другое (категория other)

Функция	Результат	Тип события	Примеры
`collect_cpu_info`	Возвращает количество событий определенного типа.	Собирает информацию о ЦП.	`dr_sandbox.descr_tech_lbcl.other.collect_cpu_info`
`collect_network_info`	Возвращает количество событий определенного типа.	Собирает информацию о сетевой активности.	`dr_sandbox.descr_tech_lbcl.other.collect_network_info`
`collect_os_info`	Возвращает количество событий определенного типа.	Собирает информацию об ОС.	`dr_sandbox.descr_tech_lbcl.other.collect_os_info`
`collect_ram_info`	Возвращает количество событий определенного типа.	Собирает информацию о RAM.	`dr_sandbox.descr_tech_lbcl.other.collect_ram_info`
`read_info_from_proc_kallsyms`	Возвращает количество событий определенного типа.	Читает информацию из /proc/kallsyms.	`dr_sandbox.descr_tech_lbcl.other.read_info_from_proc_kallsyms`

Описание функций для детектов (категория detects)

Функция	Результат	Тип события	Примеры
`all_detects_here(regexp)`	Возвращает количество событий определенного типа.	Все детекты.	`dr_sandbox.detects.all_detects_here(/Virlock/)`
`all_detects_here_num`	Возвращает количество событий определенного типа.	Все детекты.	`dr_sandbox.detects.all_detects_here_num`
`detects_of_allocs(regexp)`	Возвращает количество событий определенного типа.	Детекты файлов alloc.	`dr_sandbox.detects.detects_of_allocs(/Virlock/)`
`detects_of_allocs_num`	Возвращает количество событий определенного типа.	Детекты файлов alloc.	`dr_sandbox.detects.detects_of_allocs_num`
`detects_of_drops(regexp)`	Возвращает количество событий определенного типа.	Детекты дропов.	`dr_sandbox.detects.detects_of_drops(/Virlock/)`
`detects_of_drops_num`	Возвращает количество событий определенного типа.	Детекты дропов.	`dr_sandbox.detects.detects_of_drops_num`
`detects_of_dumps(regexp)`	Возвращает количество событий определенного типа.	Детекты дампов.	`dr_sandbox.detects.detects_of_dumps(/Virlock/)`
`detects_of_dumps_num`	Возвращает количество событий определенного типа.	Детекты дампов.	`dr_sandbox.detects.detects_of_dumps_num`
`detects_of_injects(regexp)`	Возвращает количество событий определенного типа.	Детекты инжектов.	`dr_sandbox.detects.detects_of_injects(/Virlock/)`
`detects_of_injects_num`	Возвращает количество событий	Детекты инжектов.	`dr_sandbox.detects.detects_of_injects_num`

Функция	Результат	Тип события	Примеры
	определенного типа.		
`detects_of_src(regexp)`	Возвращает количество событий определенного типа.	Детекты файлов src.	`dr_sandbox.detects.detects_of_src(/Virlock/)`
`detects_of_src_num`	Возвращает количество событий определенного типа.	Детекты файлов src.	`dr_sandbox.detects.detects_of_src_num`

Прочие функции

Функция	Описание	Примеры
`check_buffer(offset, buffer_asciihex_value)`	Проверяет буфер asciihex по заданному смещению. Длина должна быть четной. Может использоваться вместо строк, чтобы не замедлять сканирование. Возвращает 1, если строка найдена, в противном случае — 0.	`dr_sandbox.check_buffer(0,"4d5A")`
`check_byte(offset, byte_value)`	Проверяет байты по заданному смещению. Может использоваться вместо строк, чтобы не замедлять сканирование. Возвращает 1, если значение в байтах найдено, в противном случае — 0.	`dr_sandbox.check_byte(0,0x4d)`
`check_dword(offset, dword_value)`	Проверяет DWORD по заданному смещению. Может использоваться вместо строк, чтобы не замедлять сканирование. Возвращает 1, если значение DWORD	`dr_sandbox.check_dword(0,0x00905A4D)`

Функция	Описание	Примеры
	найдено, в противном случае — 0.	
`check_word(offset, word_value)`	Проверяет WORD по заданному смещению. Может использоваться вместо строк, чтобы не замедлять сканирование. Возвращает 1, если значение WORD найдено, в противном случае — 0.	`dr_sandbox.check_word(0,0x5a4d)`
`ci_any(string)`	Возвращает 1, если строка символов ASCII или wide, нечувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.ci_any("string")`
`ci_any_num(string)`	Возвращает количество найденных строк символов ASCII или wide, нечувствительных к регистру.	`dr_sandbox.ci_any_num("string")`
`ci_ascii(string)`	Возвращает 1, если строка символов ASCII, нечувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.ci_ascii("string")`
`ci_ascii_num(string)`	Возвращает количество найденных строк символов ASCII, нечувствительных к регистру.	`dr_sandbox.ci_ascii_num("string")`
`ci_wide(string)`	Возвращает 1, если строка символов wide, нечувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.ci_wide("string")`
`ci_wide_num(string)`	Возвращает количество найденных строк символов wide, нечувствительных к регистру.	`dr_sandbox.ci_wide_num("string")`
`ci_xor(string)`	Возвращает 1, если нечувствительная к	`dr_sandbox.ci_xor("string")`

Функция	Описание	Примеры
	регистру 1-байтовая строка символов ASCII, к которой применена операция XOR, найдена, в противном случае — 0.	
`ci_xor_num(string)`	Возвращает количество найденных нечувствительных к регистру 1-байтовых строк символов ASCII, к которым применена операция XOR.	`dr_sandbox.ci_xor_num("string")`
`crc32(integer, integer)`	Вычисляет и возвращает хеш crc32 для буфера. Первый параметр — смещение, второй — длина буфера.	`dr_sandbox.crc32(0, 0)`
`cs_any(string)`	Возвращает 1, если строка символов ASCII или wide, чувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.cs_any("string")`
`cs_any_num(string)`	Возвращает количество найденных строк символов ASCII или wide, чувствительных к регистру.	`dr_sandbox.cs_any_num("string")`
`cs_ascii(string)`	Возвращает 1, если строка символов ASCII, чувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.cs_ascii("string")`
`cs_ascii_num(string)`	Возвращает количество найденных строк символов ASCII, чувствительных к регистру.	`dr_sandbox.cs_ascii_num("string")`
`cs_wide(string)`	Возвращает 1, если строка символов wide, чувствительная к регистру, найдена, в противном случае — 0.	`dr_sandbox.cs_wide("string")`

Функция	Описание	Примеры
`cs_wide_num(string)`	Возвращает количество найденных строк символов wide, чувствительных к регистру.	`dr_sandbox.cs_wide_num("string")`
`detects_of_this_file(regex)`	Возвращает количество детектов для проверяемого файла.	`dr_sandbox.detects_of_this_file(/Virlock/) == 0`
`detects_of_this_file_num`	Возвращает количество детектов для проверяемого файла.	`dr_sandbox.detects_of_this_file_num`
`filename(regex)`	Возвращает 1, если регулярное выражение в имени файла найдено, в противном случае — 0.	`dr_sandbox.filename(/xtbl/)`
`filename_boost_regex(string_with_regex)`	Ищет регулярное выражение в имени файла, используя `boost::regex`. Флаги для регулярного выражения: `boost::regex::perl`. Поиск по `boost::regex_search`. Может использоваться, если нужно использовать функции регулярного выражения, которые отсутствуют в regex YARA. Например, отрицательное опережающее выражение или обратные ссылки. При этом обратите внимание, что некорректное регулярное выражение замедлит проверку. Регулярные выражения YARA работают быстрее, чем `boost::regex`, поэтому по возможности рекомендуем использовать функцию `dr_sandbox.filename(//)`. Возвращает 1, если регулярное выражение	`dr_sandbox.filename_boost_regex("(?<!abc)def")`

Функция	Описание	Примеры
	найдено, в противном случае — 0.	
`filesystem_access(regex)`	Функция высокого уровня, которая сопоставляет все операции файловой системы с регулярным выражением.	`dr_sandbox.filesystem_access(/AnnaKournikova\.jpg\.vbs/)`
`network_access(regex)`	Функция высокого уровня, которая сопоставляет все сетевые операции с регулярным выражением.	`dr_sandbox.network_access(/\.php\?id=[0-9]+&token=[0-9]+/)`
`registry_access(regex)`	Возвращает количество операций с реестром.	`dr_sandbox.registry_access(/pattern/)`
`sb_filetype`	Возвращает тип файла. Используется для сравнения со следующими константами `SB_FILETYPE_*`: `SB_FILETYPE_SRC`; `SB_FILETYPE_DROP`; `SB_FILETYPE_MEMDMP`; `SB_FILETYPE_ALLOC`; `SB_FILETYPE_DUMP`; `SB_FILETYPE_INJECT`.	`dr_sandbox.sb_filetype == dr_sandbox.SB_FILETYPE_SRC`
`search_substring_in_range(string, integer, integer)`	Выполняет поиск подстроки в буфере с использованием алгоритма Бойера — Мура. Первый аргумент — строка asciihex, второй — смещение, третий — длина. Обратите внимание, что эта функция может снижать производительность.	`dr_sandbox.search_substring_in_range("string", 0, 0)`

17. Приложение В. Настройка отдельного VPN-сервера

Ниже приведены примеры команд для CentOS. Для другой ОС необходимо использовать эквивалентные команды.

Чтобы настроить отдельный VPN-сервер

1. Перейдите в учетную запись с административным доступом к системе:

```
$ su
```

2. Установите репозиторий EPEL:

```
# yum install epel-release
```

3. Установите OpenVPN и утилиту Easy-RSA:

```
# yum install openvpn easy-rsa
```

4. Перейдите в каталог с установленной утилитой Easy-RSA:

```
# cd /usr/share/easy-rsa/3
```

5. Создайте структуру PKI-ключей:

```
# ./easyrsa init-pki
```

6. Создайте корневой удостоверяющий центр (CA):

```
# ./easyrsa build-ca
```

7. На запрос Enter New CA Key Passphrase, установите пароль на подпись сертификатов.
8. Создайте запрос сертификата для сервера без использования пароля:

```
# ./easyrsa gen-req server nopass
```

9. Подпишите запрос от CA:

```
# ./easyrsa sign-req server server
```

10. Введите пароль от CA, если он был установлен ранее.
11. Сгенерируйте ключ Диффи-Хеллмана:

```
# ./easyrsa gen-dh
```

12. Скопируйте полученные файлы в каталог с сервером OpenVPN:

```
# cp pki/ca.crt /etc/openvpn/ca.crt
```

```
# cp pki/dh.pem /etc/openvpn/dh.pem
# cp pki/issued/server.crt /etc/openvpn/server.crt
# cp pki/private/server.key /etc/openvpn/server.key
```

13. Сгенерируйте ключ для шифрования TLS Control Channel:

```
# openvpn --genkey --secret /etc/openvpn/tc.key
```

14. Создайте ключи для клиента OpenVPN-сервера:

```
# ./easyrsa gen-req vxcube nopass
# ./easyrsa sign-req client vxcube
```

15. Скопируйте ключи клиента (`vxcube.key`, `vxcube.crt`) и ключ сервера (`ca.crt`) в каталог с дистрибутивом Dr.Web vxCube (каталог `~/confs`) из следующих каталогов:

```
# /usr/share/easy-rsa/3/pki/private/vxcube.key
# /usr/share/easy-rsa/3/pki/issued/vxcube.crt
# /etc/openvpn/ca.crt
```

16. Создайте файл конфигурации OpenVPN:

```
nano /etc/openvpn/server/server.conf
```

Значения:

```
port 1194
proto udp
dev tap
ca /etc/openvpn/ca.crt
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key
dh /etc/openvpn/dh.pem
server 10.42.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
keepalive 10 120
comp-lzo
user nobody
```

```
group nobody
persist-key
persist-tun
status openvpn-status.log
log-append /var/log/openvpn.log
verb 1
```

17. Запустите сервер:

```
# systemctl start openvpn-server@server.service
```

18. Убедитесь, что сервер запущен:

```
# netstat -tulnp | grep 1194
```

19. Убедитесь, что на сервере включена переадресация IP:

```
# nano /etc/sysctl.conf
```

Добавьте строку:

```
net.ipv4.ip_forward = 1
```

20. Принудительно перечитайте конфигурацию sysctl:

```
# sysctl -p /etc/sysctl.conf
```

21. Убедитесь, что нет других сервисов, принимающих подключения от внешнего интерфейса, либо доступ к ним заблокирован настройками системного брандмауэра:

```
# netstat -nlpt
```

22. Если такие сервисы есть, добавьте правило для блокировки доступа к ним из VPN-туннеля:

```
# iptables -A INPUT -i tap0 -j DROP
```

23. Настройте маскарадинг для пакетов из подсети VPN-туннеля на интерфейс по умолчанию:

```
# iptables -t nat -A POSTROUTING -s 10.42.0.0/24 -o eth0 -j
MASQUERADE
```

18. Приложение Г. Список используемых сетевых портов

Если все компоненты находятся на одном сервере, то они взаимодействуют друг с другом локально, и потребуется открыть только порт 80 (для HTTP) и/или порт 443 (для HTTPS).

Если компоненты находятся на разных серверах, то для их взаимодействия должны быть открыты следующие сетевые порты:

vxCube Web (vxcube_web_host), входящие подключения:

- 80 (для HTTP)
- 443 (для HTTPS)
- 21 (хранилище, ТОЛЬКО для других узлов) исходящие подключения:
- hyperbox_api_host:5003

vxCube DB (hyperbox_api_host), входящие подключения:

- 25672 (RabbitMQ, взаимодействие между узлами)
- 4369 (RabbitMQ, служба обнаружения одноранговых узлов)
- 5672 (RabbitMQ, AMQP)
- 5003 (vxcube flow api app) исходящие подключения:
- vxcube_web_host:21

Windows/Android/Linux Sandbox Service (hyperbox_hosts, dimas_hosts, linuxbox_hosts), исходящие подключения:

- hyperbox_api_host:25672
- hyperbox_api_host:4369
- hyperbox_api_host:5672
- vxcube_web_host:21
- vpn_server:vpn_port (ВНЕШНИЙ, для перенаправления трафика от песочниц)
- proxy_server:proxy_port (ВНЕШНИЙ, зависит от настроек анализа)

Dr.Web Scan Service (drweb_srv_hosts), для исходящих подключений:

- hyperbox_api_host:25672
- hyperbox_api_host:4369
- hyperbox_api_host:5672
- vxcube_web_host:21
- update.geo.drweb.com:80 (ВНЕШНИЙ, для обновления антивируса)

Yara Service (yara_hosts), для исходящих подключений:

- hyperbox_api_host:25672
- hyperbox_api_host:4369
- hyperbox_api_host:5672
- vxcube_web_host:21

Analyser Service (evparser_hosts), для исходящих подключений:

- hyperbox_api_host:25672
- hyperbox_api_host:4369
- hyperbox_api_host:5672
- vxcube_web_host:21
- links-checker.dev.drweb.com:80 (ВНЕШНИЙ, опциональный, проверка доменов)