VAPORWARE

Общество с ограниченной ответственностью
«Вэйпорваре»

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «ИНФОРМАЦИОННАЯ СИСТЕМА ОЦЕНКИ РИСКОВ» (ИСОР)

Руководство по развертыванию

Версия 1

ЛИСТОВ 8

г. Санкт-Петербург
2023 г.

Подпись и дата	
Инв. № дубл.	
Взам. инв. №	
Подпись и дата	
Инв. № подл.	

АННОТАЦИЯ

Настоящий документ представляет собой руководство по развертыванию (далее – Руководство) программного обеспечения «Информационная система оценки рисков» (ИСОР, далее – ПО) и предназначен для администраторов ПО.

ПО предназначено для расчета рисков информационной безопасности в отношении активов организации и позволяет:

– формировать перечень активов организации;

– формировать перечень актуальных угроз безопасности информации активов;

– проводить количественно-качественную оценку рисков активов;

– формировать отчеты по результатам оценки рисков.

УСЛОВНЫЕ ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БД	База данных
ОС	Операционная система
ПО	Программное обеспечение
СУБД	Система управления базами данных

СОДЕРЖАНИЕ

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Комплект поставки

ПО поставляется в виде электронного дистрибутива или DVD-диска. В комплект поставки входят:

- файл стандартного образа виртуальной машины «Risk assessment.ova», совместимый с программными продуктами виртуализации Oracle VM VirtualBox, VMware vSphere, Hyper-V;
- файл формата дискового образа программы QEMU «Risk-assessment-disk001.qcow2» для использования в программном продукте виртуализации KVM;
- документ «[ИСОР] Руководство по развертыванию.pdf»;
- документ «[ИСОР] Руководство пользователя.pdf»;
- документ «[ИСОР] Сертификат технической поддержки.pdf».

В отдельных случаях предоставляются репозитории для развертывания на уже введенных в эксплуатацию операционных системах:

- репозиторий «ara-backend-master»;
- репозиторий «ara-frontend-master».

1.2 Системные требования

Используются следующие характеристики образов виртуальных машин:

- Процессор – 4 ядра,
- Оперативная память – 4096 МБ,
- Дисковое пространство – 10 ГБ,
- ОС Linux (Ubuntu 18.04),
- СУБД MongoDB Community Edition версии 4.2 или выше,
- Node.js версии 12.16.3 (или 16.13.0 LTS),
- NPM версии 6.14.4 (или 8.1.0 - для версии 16.13.0 LTS).

1.3 Учетные данные по умолчанию

Для входа в ОС Linux при использовании образов «Risk assessment.ova» и «Risk-assessment-disk001.qcow2» используются следующие учетные данные по умолчанию:

Логин: risk

Пароль: 78ZD783vcN2bfzqP

Для работы с СУБД MongoDB Community Edition при использовании образов «Risk assessment.ova» и «Risk-assessment-disk001.qcow2» используются следующие учетные данные по умолчанию:

Логин: root

Пароль: 78ZD783vcN2bfzqP

Для входа в интерфейс ПО используются следующие учетные данные по умолчанию:

Учетная запись администратора: admin

Пароль: name32

2 АДМИНИСТРАТИВНЫЕ ОПЕРАЦИИ

2.1 Развертывание образов виртуальной машины

Порядок монтирования образа осуществляется в соответствии с документацией на используемый программный продукт виртуализации.

Сетевой интерфейс виртуальной машины необходимо настроить на режим «Сетевой мост» (рисунок 2.1).

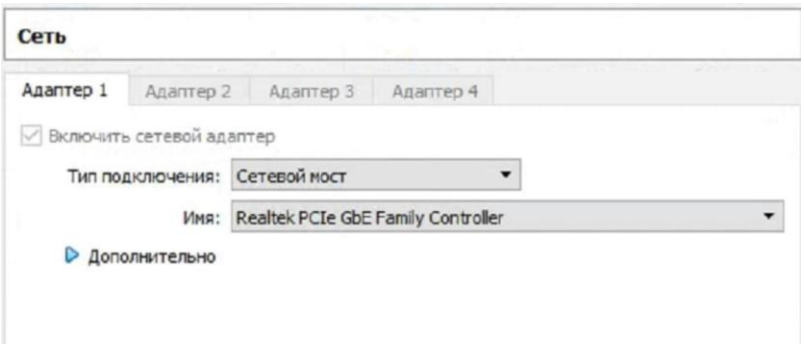

Рисунок 2.1 – Режим «Сетевой мост» в Oracle VM VirtualBox

После запуска виртуальной машины необходимо осуществить вход в операционную систему Linux с использованием учетных данных администратора по умолчанию (см. раздел 1.3).

Для того, чтобы узнать ip-адрес интерфейса ПО необходимо ввести команду *ip addr show.*

Доступ в интерфейс ПО осуществляется по полученному в результате вывода команды ip-адресу через браузер по адресу http://ip-адрес.

2.2 Развертывание репозиториев

В зависимости от заключенного лицензионного договора возможно предоставление репозиториев ПО. Установка репозиториев выполняется следующим образом:

1. Установить ОС Linux (Ubuntu 18.04).
2. Установить СУБД MongoDB Community Edition версии 4.2 или выше.
3. Запустить MongoDB

sudo systemctl start mongod

4. Подключиться к консоли СУБД

mongod --port 27017 (если база была запущена с портом по умолчанию)

5. Перейти в БД *admin*

use admin

6. Создать пользователя СУБД с помощью команды

db.createUser() с ролью *root* в БД *admin* (см. документацию)

Для рабочей инфраструктуры рекомендуется следовать рекомендациям по безопасности MongoDB.

7. Установить Node.js версии 12.16.3 (или 16.13.0 LTS).
8. Установить NPM версии 6.14.4 (или же 8.1.0 - для версии 16.13.0 LTS)
9. В директории проекта *ara-backend* отредактировать файл *config.json*: изменить строку подключения к экземпляру MongoDB (параметр mongoDB.uri) - указать параметры

подключения (для тестирования используем ранее созданного root-пользователя).

10. Перейти в директорию *certs*, сгенерировать ключ и сертификат для JWT-аутентификации, а затем указать путь до них в *config.json*. Параметры: для ключа - *jwt.key*, для сертификата - *jwt.cert.*

Пример генерации:

openssl genrsa -out server-key.pem 2048

openssl req -new -sha256 -key server-key.pem -out server-csr.pem

openssl x509 -req -in server-csr.pem -signkey server-key.pem -out server-cert.pem

Соответственно,

путь для ключа - *jwt.key = "./certs/server-key.pem"*,

для сертификата *jwt.cert = "./certs/server-cert.pem"*

11. Находясь в корневой директории проекта *ara-backend*, установить библиотеки с помощью команды:

npm i

12. Находясь в корневой директории *ara-backend*, выполнить команду:

node init.js

Скрипт создаст пользователя-администратора *admin* с паролем *name32*.

С его помощью можно начать работу с системой.

13. Запустить проект *ara-backend* командой:

node index.js

14. Находясь в корневой директории проекта *ara-frontend*, установить библиотеки с помощью команды:

npm i

15. Запустить сборку проекта *ara-frontend* командой:

npm run build

16. Сконфигурировать инфраструктуру для доступа к статическим файлам проекта *ara-frontend* в папке *build*, а также проксирование запросов к порту 8080 проекта *ara-backend*.

2.3 Конвертация образа OVA в KVM

1. Скопируйте на хост виртуальную машину в формате *.ova.

2. OVA – это архив, из которого необходимо извлечь файл жесткого диска в формате *.vmdk (остальные файлы можно удалить):

root@debian:~# tar -xvf /tmp/<file_name>.ova –C /tmp

*.vmdk и *.img форматы имеют ряд ограничений при использовании в KVM, поэтому рекомендуется использовать формат qcow2.

3. Необходимо конвертировать файл формата *.vmdk в файл формата *.qcow2.

root@debian:~# qemu-img convert -f vmdk -O qcow2 /tmp/<file_name>.vmdk /home/<file_name>.qcow2

4. Создайте виртуальную машину:

root@debian:~# virt-install --connect=qemu:///system -n <system_name> -r 4096 --vcpus=4 --import

--disk path=/home/<file_name>.qcow2,format=qcow2,bus=virtio --vnc --noautoconsole --os-type=linux --accelerate --network=bridge:br0,model=virtio

Где:

--connect=qemu:///system – URL, по которому происходит подключение к KVM;

-n <имя> – название виртуальной машины;

-r <количество> – количество памяти в МБ, выделенных для виртуальной машины;

--vcpus=<количество> – количество процессорных ядер, выделенных для виртуальной машины;

--import – использовать для виртуальной машины имеющийся образ диска;

--disk path=<путь до образа диска> – путь до диска (совместное использование с --import указывает на существующий диск); format=<формат образа>,bus=virtio – обязательные параметры для образов vmdk и qcow2;

--os-type=<ОС> – тип операционной системы виртуальной машины;

--vnc – запуск vnc для доступа к консоли, без этой команды через команду virsh console будет не подключиться;

--noautoconsole – не пытаться автоматически подсоединиться к консоли после создания виртуальной машины;

--accelerate – работа через /dev/kvm;

--network=bridge:<название интерфейса> – создание сетевого адаптера с привязкой к конкретному бриджу.