ООО «ТИОНИКС»

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «TIONIX VIRTUAL SECURITY»

ИНСТРУКЦИЯ ПО УСТАНОВКЕ

RU.НРФЛ.00002-01.96.01

ЛИСТОВ 10

МОСКВА, 2020

Данная инструкция применима для установки ПО TIONIX Virtual Security на CentOS 7 x86_64 (minimal install).

1. Описание состава дистрибутива

Дистрибутив состоит из следующих пакетов (Версия пакетов может отличаться в зависимости от версии дистрибутива, в данном руководстве версия 1.0.0):

	Наименование	**Имя файла пакета**	**Минимальные требования к RAM**	**Описание**
1	Скрипты конфигурации	tionix-tvs-configure-1.0.0-el7.x86_64.rpm	-	Пакет, содержащий в себе скрипты конфигурации системы
2	Виртуальной машины Java	tionix-tvs-openjdk-11.0.7_10-el7.x86_64.rpm	-	Пакет, содержащий в себе сборку виртуальной машины, необходимый для функционирования всех модулей продукта
3	Пакет базы данных	tionix-tvs-db-1.0.0-el7.x86_64.rpm	1 GB	Мета-пакет, который содержит в себе зависимость на базу данных PostgreSQL 12, зависит от (1)
4	Модуль безопасности	tionix-tvs-security-1.0.0-el7.x86_64.rpm	2 GB	Пакет, содержащий модуль сервера безопасности, зависит от (1,2)
5	Модуль администрирования (ядро системы)	tionix-tvs-core-1.0.0-el7.x86_64.rpm	2 GB	Пакет, содержащий модуль администрирования, зависит от (1,2)
6	Веб-сервер администрирования	tionix-tvs-web-1.0.0-el7.x86_64.rpm	1GB	Пакет, содержащий веб сервер администрирования, зависит от (1,2)

7	Сервер балансировки нагрузки	tionix-tvs-balancer-1.0.0-el7.x86_64.rpm	1GB	Пакет, содержащий сервер балансировки нагрузки, зависит от (1,2)
8	Агент вычислительного узла	tionix-tvs-agent-1.0.0-el7.x86_64.rpm	512MB	Пакет, содержащий в себе агент, устанавливаемый на вычислительный узел, зависит от (1,2)
9	Пакет системы без вычислительного узла	tionix-tvs-system-1.0.0-el7.x86_64.rpm	7 GB	Мета-пакет, для установки системы целиком без агента вычислительного узла, зависит от (3,4,5,6,7). Используется для развертки всего дистрибутива на одной операционной системе, за исключением вычислительных узлов.
10	Пакет системы целиком	tionix-tvs-all-1.0.0-el7.x86_64.rpm	7.5 GB	Мета-пакет, для установки системы целиком без агента вычислительного узла, зависит от (3,4,5,6,7) или (9). Используется для развертки всего дистрибутива на одной операционной системе.

- Пакет базы данных (2) зависит от сервера PostgreSQL 12., который включает в себя следующие пакеты:

Имя файла	**Описание**
postgresql12-libs-12.2-2PGDG.rhel7.x86_64.rpm	Библиотеки PostgreSQL
postgresql12-12.2-2PGDG.rhel7.x86_64.rpm	Клиент PostgreSQL
postgresql12-server-12.2-2PGDG.rhel7.x86_64.rpm	Сервер PostgreSQL
libseccomp-2.3.1-3.el7.x86_64.rpm	Вспомогательная библиотека
libicu-50.2-3.el7.x86_64.rpm	Вспомогательная библиотека

Данные зависимости можно получить автоматически при наличии интернета на целевой машине:

- с помощью команды установки пакета:

```
yum install https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

- с помощью скрипта **download_deps.sh** на машине с интернетом и дальнейшим переносом их на целевую машину. После запуска скрипта пакеты будут скачаны в каталог **postgresql.**

- Пакет агента вычислительного узла, зависит от библиотеки libvirt, а также модулей ldap. При наличии интернета на целевой машине пакеты будут загружены автоматически. В случае отсутствия интернета зависимости можно загрузить с помощью скрипта **download_deps.sh** на машине с интернетом, перенеся их на целевую машину. После запуска скрипта пакеты будут скачаны в каталоги **agent** и **ldap.**
- Остальные пакеты системы (3,4,5,6,7) зависят от пакета синхронизации времени, который будет установлен автоматически при наличии интернета на целевой машине. В случае отсутствия интернета зависимости можно загрузить с помощью скрипта **download_deps.sh** на машине с интернетом, перенеся их на целевую машину. После запуска скрипта пакеты будут скачаны в каталог **system**.

2. Варианты развертки

- В минимальной конфигурации, вся система может быть установлена на одну машину с помощью установки пакета (10).
- В конфигурации разделения вычислительных узлов, и системы необходимо установить систему на одну машину с помощью пакета (9), а на машины вычислительных узлов установить пакет (8).
- В максимально развернутой конфигурации, пакеты (4, 5, 6) - могут быть установлены каждый на отдельную машину, и могут быть продублированы более чем на одну машину. Агенты (8) - каждый устанавливается по одному на вычислительный узел, а сервер балансировки нагрузки (7) может быть установлен на одну или несколько машин при возможности внешней балансировки на них, например, через DNS или NgInx.

Все машины для установки должны быть в одной подсети. Также необходимо учитывать минимальные требования к RAM каждого модуля. При развертке системы, для удобства ее администрирования и настройки желательно, но необязательно, чтобы все машины имели доступ друг к другу по имени, что обеспечивается либо через DNS, либо записями в файл /etc/hosts каждой машины имен всех остальных.

3. Пример установки в изолированном окружении

Пример установки системы в конфигурации "Все в одном" в **изолированном** окружении, на машину с адресом **192.168.200.150** с DNS именем **tvsdemo.ru , RAM 16GB**:

Получаем зависимости на машине с доступом в интернет, и копируем на целевую машину:

```
$ ./download_deps.sh

$ scp -pr postgres root@192.168.200.150:/root/deps
$ scp -pr system root@192.168.200.150:/root/deps
$ scp -pr agent root@192.168.200.150:/root/deps
$ scp -pr ldap root@192.168.200.150:/root/deps
$ scp tionix-tvs-*.rpm root@192.168.200.150:/root/

tionix-tvs-agent-1.0.0-el7.x86_64.rpm
tionix-tvs-all-1.0.0-el7.x86_64.rpm
tionix-tvs-balancer-1.0.0-el7.x86_64.rpm
tionix-tvs-configure-1.0.0-el7.x86_64.rpm
tionix-tvs-core-1.0.0-el7.x86_64.rpm
tionix-tvs-db-1.0.0-el7.x86_64.rpm
tionix-tvs-openjdk-11.0.7_10-el7.x86_64.rpm
tionix-tvs-security-1.0.0-el7.x86_64.rpm
tionix-tvs-system-1.0.0-el7.x86_64.rpm
tionix-tvs-web-1.0.0-el7.x86_64.rpm
```

$

Устанавливаем пакеты на целевой машине:

```
ssh root@192.168.200.150
root@192.168.200.150's password:

[root@localhost ~]#  yum --disablerepo=* install ./tionix-tvs-*
deps/system/* deps/agent/* deps/ldap/* deps/*.rpm
....
Установить  198 пакетов
Обновить      9 пакетов

Общий размер: 1.3 G
Is this ok [y/d/N]: y

....
Выполнено!
```

Выполняем конфигурацию системы:

```
[root@localhost ~]# tvs_configure.sh
Укажите сеть доступа к серверу PostgreSQL (пример, 192.168.0.0/24):
192.168.200.0/24 #указывается подсеть, в которой работает система
Укажите IP-адрес сервера базы данных (пример, 192.168.0.1): 192.168.200.150
#указывается IP адрес, на котором будет слушать сокет сервер PostgreSQL
Укажите имя базы данных [tvs]: #по умолчанию tvs
Введите пользователя базы данных [tvs]:  #по умолчанию tvs
Введите пароль пользователя базы данных [tvs]: #по умолчанию tvs
Повторите ввод пароля [tvs]:  #по умолчанию tvs
Настройка PostgreSQL...
Initializing database ... OK

Запуск PostgreSQL...
Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql-
12.service to /usr/lib/systemd/system/postgresql-12.service.
```

Настройка базы данных...
CREATE ROLE
CREATE DATABASE
CREATE SCHEMA
CREATE SCHEMA
Сохранение конфигурации...
Укажите имя узла сервера балансировки (в кластере не должно быть одинаковых имен) [localhost.localdomain-balancer]: **#по умолчанию как hostname+"-balancer", в среде с несколькими машинами нужно задавать различные имена**
Укажите IP-адрес для сервера балансировки, он должен быть доступен для всех узлов системы (пример, 192.168.0.1): 192.168.200.150 **#адрес, на котором слушает сервер балансировки**
Укажите IP-адрес внешнего свервера времени (пустой ввод - будут использованы сервера по умолчанию) []: **#по умолчанию сервера CentOS, но можно указать любой внутренний сервер NTP**
Настройка сервера балансировки...
net.core.wmem_max = 1048576
net.core.rmem_max = 26214400
Сохранение конфигурации...
Укажите имя узла сервера безопасности (в кластере не должно быть одинаковых имен) [localhost.localdomain-security]: **#по умолчанию как hostname+"-security",в среде с несколькими машинами нужно задавать различные имена**
Укажите IP-адрес для сервера безопасности (пример, 192.168.0.1): 192.168.200.150 **#ip-адрес, на котором будет работать сервер безопасности**
Настройка сервера безопасности...
net.core.wmem_max = 1048576
net.core.rmem_max = 26214400
Запуск сервера безопасности...
Сохранение конфигурации...
Укажите имя узла сервера администрирования (в кластере не должно быть одинаковых имен) [localhost.localdomain-core]: **#по умолчанию как hostname+"-security",в среде с несколькими машинами нужно задавать различные имена**
Укажите IP-адрес для сервера администрирования (пример, 192.168.0.1) [192.168.200.150]: **#ip-адрес, на котором будет работать сервер администрирования**
Укажите URL-системы (пример, https://my.domain.ru:8080): : http://tvsdemo.ru:8080 **#Адрес, по которому будет доступна система пользователям, обратите внимание на порт, необходимо указать 8080**
Укажите IP-адрес системы (пример, 192.168.0.1): : 192.168.200.150 **#ip-адрес, который соответствует доменному имени tvsdemo.ru, и по нему же должна быть доступна система.**
Настройка сервера администрирования...
net.core.wmem_max = 1048576
net.core.rmem_max = 26214400
Запуск сервера администрирования...
Сохранение конфигурации...
Укажите имя узла web сервера (в кластере не должно быть одинаковых имен) [localhost.localdomain-web]: **#по умолчанию как hostname+"-web",в среде с несколькими машинами нужно задавать различные имена**
Укажите IP-адрес для web сервера (пример, 192.168.0.1) [192.168.200.150]: **#ip-адрес, на котором будет работать web сервер**
Файл hosts уже содержит запись для tvsdemo.ru, заменить (y/[n])? n
Настройка web сервера...
net.core.wmem_max = 1048576
net.core.rmem_max = 26214400
Запуск web сервера, это может занять продолжительное время...
Сохранение конфигурации...

```
Укажите идентификатор домена безопасности [master]: #идентификатор домена системы, по умолчанию master
Укажите идентификатор вычислительного узла (GUID) [42a0bdd9-dc1d-4498-abf2-340a124649a3]: #идентификатор вычислительного узла
Укажите через запятую адреса или имена серверов безопасности (пример: 192.168.0.1,192.168.0.2 или security-1,security2): 192.168.200.150 #адреса узлов серверов безопасности
Настройка агента безопасности...
Запуск агента безопасности, это может занять продолжительное время...
Сохранение конфигурации...
[root@localhost ~]#
```

Для доступа к системе открываем ее страницу http://tvsdemo.ru:8080 в браузере:

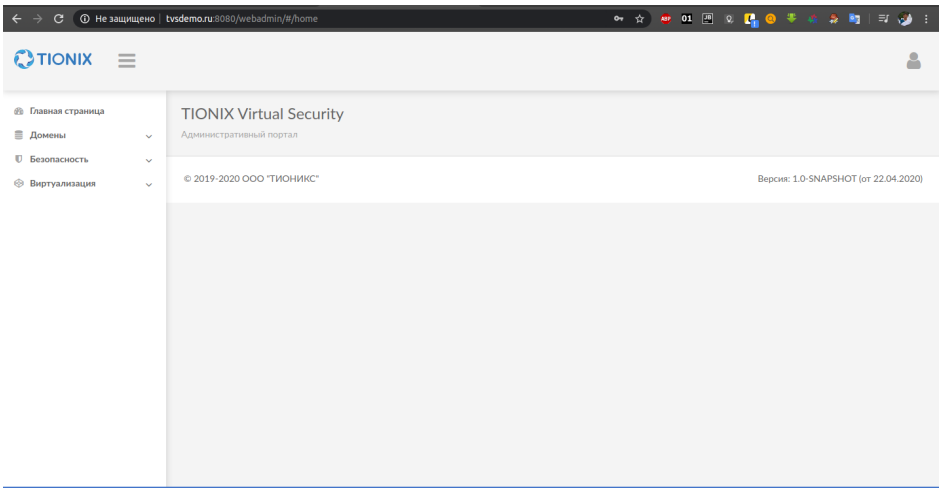

Рис. 1 Главная страница интерфейса

4. Пример установки в среде с доступом в интернет

Система будет развернута на три машины:

192.168.122.121 - сервер БД (RAM 4 GB)
192.168.122.4 - система (RAM 10 GB)
192.168.122.166 - вычислительный узел (RAM 2 GB)

4.1 Развертка сервера БД

Копируем пакеты на целевую машину:

```
$ scp tionix-tvs-configure-1.0.0-el7.x86_64.rpm root@192.168.122.121:/root/
$ scp tionix-tvs-db-1.0.0-el7.x86_64.rpm root@192.168.122.121:/root/
```

Устанавливаем пакеты на целевой машине:

```
$ ssh root@192.168.122.121
yum install -y  https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-
x86_64/pgdg-redhat-repo-latest.noarch.rpm
yum install -y tionix-tvs-configure-1.0.0-el7.x86_64.rpm tionix-tvs-db-
1.0.0-el7.x86_64.rpm
```

Настраиваем сервер БД:

```
tvs_configure.sh
Укажите сеть доступа к серверу PostgreSQL (пример, 192.168.0.0/24):
192.168.122.0/24
Укажите IP-адрес сервера базы данных (пример, 192.168.0.1): 192.168.122.121
Укажите имя базы данных [tvs]:
Введите пользователя базы данных [tvs]:
Введите пароль пользователя базы данных [tvs]:
Повторите ввод пароля [tvs]:
Настройка PostgreSQL...
Initializing database ... OK

Запуск PostgreSQL...
Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql-
12.service to /usr/lib/systemd/system/postgresql-12.service.
Настройка базы данных...
CREATE ROLE
CREATE DATABASE
CREATE SCHEMA
CREATE SCHEMA
Сохранение конфигурации...
```

4.2 Развертка сервера Системы

Копируем пакеты на целевую машину:

```
$ scp tionix-tvs-configure-1.0.0-el7.x86_64.rpm  root@192.168.122.4:/root/
$ scp tionix-tvs-openjdk-11.0.7_10-el7.x86_64.rpm root@192.168.122.4:/root/
$ scp tionix-tvs-balancer-1.0.0-el7.x86_64.rpm root@192.168.122.4:/root/
$ scp tionix-tvs-core-1.0.0-el7.x86_64.rpm  root@192.168.122.4:/root/
$ scp tionix-tvs-web-1.0.0-el7.x86_64.rpm root@192.168.122.4:/root/
$ scp tionix-tvs-security-1.0.0-el7.x86_64.rpm  root@192.168.122.4:/root/
```

Устанавливаем пакеты на целевой машине:

```
$ ssh root@192.168.122.4
yum install -y tionix-tvs-*
```

Настраиваем сервер Системы:

```
tvs_configure.sh
Укажите имя узла сервера балансировки (в кластере не должно быть одинаковых
имен) [192.168.122.4-balancer]: balancer
Укажите IP-адрес для сервера балансировки (пример, 192.168.0.1):
192.168.122.4
Укажите IP-адрес внешнего свервера времени (пустой ввод - будут использованы
сервера по умолчанию) []:
Настройка сервера балансировки...
Запуск сервера балансировки, это может занять продолжительное время...
Сохранение конфигурации...
Укажите имя узла сервера безопасности (в кластере не должно быть одинаковых
имен) [192.168.122.4-security]: security
Укажите IP-адрес для сервера безопасности (пример, 192.168.0.1):
192.168.122.4
Укажите IP-адрес сервера базы данных (пример, 192.168.0.1): 192.168.122.121
Укажите имя базы данных [tvs]:
Введите пользователя базы данных [tvs]:
Введите пароль пользователя базы данных [tvs]:
Повторите ввод пароля [tvs]:
Настройка сервера безопасности...
Запуск сервера безопасности...
Сохранение конфигурации...
Укажите имя узла сервера администрирования (в кластере не должно быть
одинаковых имен) [192.168.122.4-core]: core
Укажите IP-адрес для сервера администрирования (пример, 192.168.0.1)
[192.168.122.4]:
Укажите URL-системы (пример, https://my.domain.ru:8080): :
http://192.168.122.4:8080
Настройка сервера администрирования...
Запуск сервера администрирования...
Сохранение конфигурации...
Укажите имя узла web сервера (в кластере не должно быть одинаковых имен)
[192.168.122.4-web]: web
Укажите IP-адрес для web сервера (пример, 192.168.0.1) [192.168.122.4]:
Настройка web сервера...
Запуск web сервера, это может занять продолжительное время...
Сохранение конфигурации...
```

4.3 Развертка вычислительного узла

Копируем пакеты на целевую машину:

```
$ scp tionix-tvs-configure-1.0.0-el7.x86_64.rpm
root@192.168.122.166:/root/
$ scp tionix-tvs-openjdk-11.0.7_10-el7.x86_64.rpm
root@192.168.122.166:/root/
$ cp tionix-tvs-agent-1.0.0-el7.x86_64.rpm root@192.168.122.166:/root/
```

Устанавливаем пакеты на целевой машине:

```
$ ssh root@192.168.122.166
yum install -y tionix-tvs-*
```

Настраиваем сервер Системы:

```
tvs_configure.sh
Укажите идентификатор домена безопасности [master]:
Укажите идентификатор вычислительного узла (GUID) [adb83d17-4618-445e-82f3-
51667338d447]:
Укажите через запятую адреса или имена серверов безопасности (пример:
192.168.0.1,192.168.0.2 или security-1,security2): 192.168.122.4
Настройка агента безопасности...
Запуск агента безопасности, это может занять продолжительное время...
Сохранение конфигурации...
```

После установки система будет доступна по адресу: http://192.168.122.4:8080/

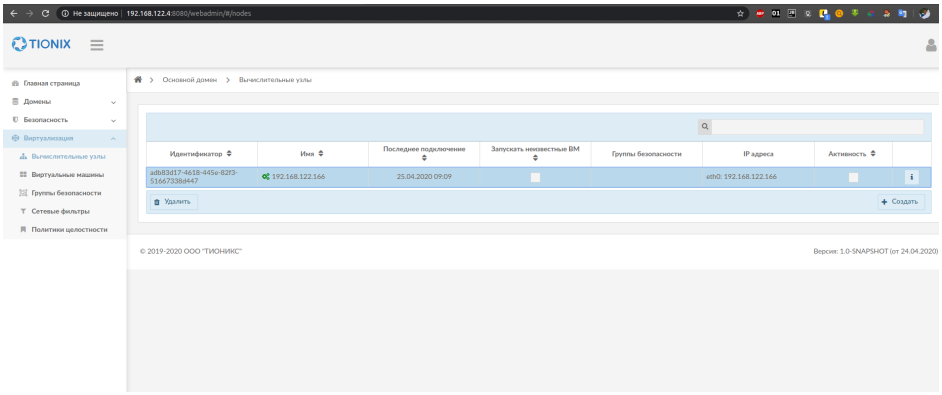

Рис. 2 Страница интерфейса. Основной домен/Вычислительные узлы

Конфигурация каждой подсистемы (размер кучи, другие параметры JVM) располагаются в файлах **/etc/tvs/<подсистема>/<подсистема>.conf**

Для подсистем безопасности, ядра, балансировщика, веб-сервера и агента создаются соответствующие **systemd** службы: **tvs-security, tvs-core, tvs-balancer, tvs-web, tvs-agent**

Журналы событий располагаются в в файлах **/var/log/tvs/<подсистема>.conf** Параметры агента вычислительного узла расположены в файле **/opt/tvs/agent/config/application.properties**

Настройки конфигурации подсистем расположены в файле: **/opt/tvs/configure/tvs_config.conf,** данный файл используется скриптом конфигурации **tvs_configure.sh**, если вы хотите переконфигурировать систему заново - либо удалите файл либо вызовите скрипт **tvs_configure** с параметром **skip_read_config**