

H F Labs

«Маскировщик»

Руководство по развертыванию

Оглавление

1 Требования к промышленной программно-аппаратной платформе.....	3
1.2 Рабочее место дата-стюарда (клиентская часть).....	3
1.3 Доступы и права	3
2 Требования к настройке программно-аппаратной платформы.....	3
2.1 Настройка рабочей станции для HFLabs.....	3
3. Настройка и запуск модуля Маскировщик	4
4. Настройка и запуск dama-service	6
Шаг 0. Скачать нужные для разворачивания сервиса файлы.....	6
Шаг 1. Настройка окружения для dama-service.....	6
Шаг 2. Настройка конфигурации сервиса	7
Шаг 3. Запуск сервиса.....	8
Шаг 4. Старт сессии	8
5. Контакты технических специалистов	9

1 Требования к промышленной программно-аппаратной платформе

Рекомендуемое количество серверов зависит от выбранного варианта решения и требований к скорости маскирования.

Для каждой ноды установки продукта:

- Процессор Intel(R) Xeon(R) Silver 4114 или выше – 16 CPU;
- 64 Гб оперативной памяти;
- SSD-диск объемом 500 Гб;
- ALT Linux 10;
- Рекомендуемые операционные системы: CentOS 7+ или Red Hat Enterprise Linux 7+, x64.
- Java SE Development Kit (OpenJDK) 11, с установленными актуальными обновлениями.

1.2 Рабочее место дата-стюарда (клиентская часть)

Для работы дата-стюардов рекомендуются клиентские машины.

- Процессор Intel Core i3 или новее;
- Оперативная память 4 Гб;
- Свободное место на жёстком диске 10 Гб;
- Сетевая карта 100 Мбит;
- Операционная система Windows 7 и выше;
- Разрядность ОС 64-bit;
- Рекомендуемый браузер: Mozilla Firefox версии 63+ или Google Chrome версии 100+

1.3 Доступы и права

1. Рабочие станции внесены в домен.
2. Создан пользователь с правами локального администратора.
3. Доступна возможность копирования файлов на сервер приложений «Маскировщика» (по RDP или иным способом).
4. Открыты доступы к СУБД источникам и приемникам с сервера, где установлен «Маскировщик»
5. Доступ к ресурсам HFLabs через сеть Интернет – confluence и jira.

2 Требования к настройке программно-аппаратной платформы

2.1 Настройка рабочей станции для HFLabs

ОС и программное обеспечение

- Windows 7 и выше;
- Java SE Development Kit (OpenJDK) 11, с установленными актуальными обновлениями;
- SQL Developer или SQL Workbench/J;
- Notepad++;
- Far Manager;
- Базовый набор утилит из набора CygWIN— ls, cat, pwd, sed, grep, awk, bash, scp, ssh;
- WinSCP;
- SoapUI;

- Firefox Quantum.

3. Настройка и запуск модуля Маскировщик

1. Создать пользователя (при необходимости) от имени которого будет производиться запуск.

```
# groupadd masker  
# useradd masker -g masker
```

2. Создать каталог для приложения.

```
# mkdir -p /opt/masker/hmt/data  
# mkdir -p /opt/masker/hmt/java
```

3. Скачиваем все необходимые файлы в папку **/ОПТ/МАСКЕР/НМТ**. **СВЕЖУЮ ВЕРСИЮ МОЖНО ВЗЯТЬ В [ТС](#)**.

a) amazon-corretto-17-x64-linux-jdk.tar.gz
б) gar_xml.zip
в) application.properties
г) hfl-masking-tool.jar

4. Распаковываем **JDК**.

```
# cd /opt/masker/hmt  
# tar xf amazon-corretto-17-x64-linux-jdk.tar.gz -C java --strip-components=1
```

5. Задаем нужные параметры в файле **APPLICATION.PROPERTIES**.

server.port - порт приложения
app.factor.url - URL до фактора
store.path - путь до папки data (/opt/masker/hmt/data)
app.gar.poolSize - количество потоков для индексации ГАР. Выставляется в зависимости от окружение, в общем случае количество ядер - 1
app.gar.source.path - путь к архиву ГАР (/opt/masker/hmt/gar_xml.zip)

6. Задаем пользователя.

```
# chown -R masker:masker /opt/masker
```

7. Создаем сервис.

Создаем файл `hmt.service` в папке `/etc/systemd/system/`

`[Unit]`

`Description=HFLabs Masking Tool`

`After=syslog.target network.target local-fs.target`

`Before=httpd.service`

`[Service]`

`User=masker`

`Group=masker`

`Type=simple`

`ExecStart=/opt/masker/hmt/java/bin/java -jar /opt/masker/hmt/hfl-masking-tool.jar`

`StandardOutput=syslog`

`StandardError=syslog`

`OOMScoreAdjust=-1000`

`WorkingDirectory=/opt/masker/hmt`

`ExecStop=/bin/kill -15 $MAINPID`

`SuccessExitStatus=143`

`[Install]`

`WantedBy=multi-user.target`

8. Перезагружаем информацию о сервисах.

```
# systemctl daemon-reload
```

9. Запускаем.

```
# systemctl start hmt
```

4. Настройка и запуск dama-service

Шаг 0. Скачать нужные для разворачивания сервиса файлы.

Все необходимые файлы выложены на FS.

Комплект поставки для dama-service состоит из:

1. dama-service-*.jar - JAR файл приложения
2. application.properties - файл настроек для приложения
3. dama.service - файл сервиса для приложения.
4. run.sh - файл запуска приложения.

Последние версии файлов **APPLICATION.PROPERTIES**, **DAMA.SERVICE**, **RUN.SH** можно найти в репозитории проекта в папке deploy. Выкладываются на FS в ручную, так как операция единократная только для первичного развертывания сервиса.

1. На сервере с приложением создать папку для файлов, которые нужны далее в процессе разворачивания и настройке сервиса.

```
sudo mkdir /opt/dama/install/
```

2. Скачать с FS необходимые файлы и переместить их в директорию /opt/dama/install/:

- вспомогательные файлы: dama.zip;
- JAR файл приложения: dama-service-*.jar.

3. Распаковать скачанный архив

```
sudo unzip /opt/dama/install/dama.zip -d /opt/dama/install/
```

Шаг 1. Настройка окружения для dama-service

1. Создать пользователя dama:

```
sudo useradd dama -g masker
```

2. Создать рабочую директорию для сервиса

```
sudo mkdir -p /opt/dama/appserver
```

3. Положить вспомогательный файлы в рабочую директорию (/OPT/DAMA/APPSERVER/):

```
sudo cp /opt/dama/install/application.properties /opt/dama/appserver/  
sudo cp /opt/dama/install/run.sh /opt/dama/appserver/
```

4. Скопировать jar-файл (в указанной ниже команды вместо * будет указана версия сервиса):

```
sudo cp /opt/dama/install/dama-service-*.jar /opt/dama/appserver/
```

5. Дать права скрипту **RUN.SH** на запуск

```
sudo chmod +x /opt/dama/appserver/run.sh
```

6. Создать вспомогательные директории для сервиса

```
sudo mkdir /opt/dama/log/  
sudo mkdir /opt/dama/h2/  
sudo mkdir /opt/dama/session/
```

7. Дать права пользователю **dama** для работы с директорией сервиса

```
sudo chown -R dama:masker /opt/dama/
```

8. Скопировать сервис файл **DAMA.SERVICE** в системную директорию сервисов

```
sudo cp /opt/dama/install/dama/dama-service/dama.service /etc/systemd/system/
```

9. Перезагрузить информацию о сервисах

```
sudo systemctl daemon-reload
```

Шаг 2. Настройка конфигурации сервиса

1. Открыть файл настроек сервиса **dama**:

```
sudo nano /opt/dama/appserver/application.properties
```

2. Установить требуемое значение порта, на котором будет развёрнут сервис **dama** следующим параметром:

```
server.port={port}
```

3. Установить URL маскировщика следующим параметром:

```
dama.masker.baseUrl={masker-url}
```

4. Установить путь к директории для конфигурационных файлов старта сессии следующим параметром:

```
dama.session.scheduled.start.directory={session-path}
```

5. Если путь до этой директории не менялся, то необходимо установить:

```
dama.session.scheduled.start.directory=opt/dama/session/
```

6. Установить путь к директории хранения логов следующим параметром:

```
logging.path={log-path}
```

7. Если путь до этой директории не менялся, то необходимо установить:

```
logging.path=/opt/dama/log/
```

8. Установить путь к директории хранения файлов in-memory базы данных H2:

```
spring.datasource.url=jdbc:h2:file:{h2-path}h2db;MODE=LEGACY
```

9. Если путь до этой директории не менялся, то необходимо установить:

```
spring.datasource.url=jdbc:h2:file:/opt/dama/h2/h2db;MODE=LEGACY
```

10. Сохранить изменения в файле (CRTL+O), подтвердить сохранение (Enter) и выйти из окна GNU nano (CTRL + X)

11. Открыть файл настроек linux сервиса для dama:

```
sudo nano /etc/systemd/system/dama.service
```

12. Установить путь к директории с sh-скриптом для старта сервиса:

```
ExecStart=/bin/bash {run.sh-path} start
```

13. Если директория создавалась без изменений на предыдущих шагах, то путь будет такой:

```
ExecStart=/bin/bash /opt/dama/appserver/run.sh start
```

14. Сохранить изменения в файле (CRTL+O), подтвердить сохранение (Enter) и выйти из окна GNU nano (CTRL + X)

Шаг 3. Запуск сервиса

1. Запустить сервис dama:

```
sudo service dama start
```

2. Проверить статус сервиса можно следующей командой:

```
sudo service dama status
```

Шаг 4. Старт сессии

1. Подготовить конфигурационный файл с описанием всего процесса маскирования:
 1. Источник данных
 2. Этапы обработки данных
 3. Приёмник данных
2. Готовый конфигурационный файл отправить REST запросом на контролер `session_controller` к методу `/session/execute` в соответствии со сваггером сервиса, который доступен по:

```
{dama-host}:{dama-port}/swagger-ui/
```


5. Контакты технических специалистов

Контакты технических специалистов, которые могут проконсультировать по процессу развёртывания и настройки экземпляра ПО и его функционирования:

- Кирилл Никитин, kirillnik@hflabs.ru
- Антон Потапенко, antonpot@hflabs.ru
- Александр Беслик, alexanderb@hflabs.ru