

Установка JIRA на Debian Lenny

Установка [JIRA](#) на Debian Lenny. Система конфигурировалась "на посмотреть", по-этому ни о какой надежности и резервном копировании речь не шла. В описанной конфигурации использовалась [HSQLDB](#), которая идет в поставке, но в "production" ее использовать не рекомендуется. Несмотря на [внятную документацию](#), дело оказалось очень непростым. В первую очередь из-за того, что система требует Java, а она к Debian прикручивается достаточно нетривиально. К тому же, в Debian (точнее, в пакете java-packge) есть ошибка, которая не дает сконвертировать бинарный пакет Java, полученный с сайта [java.sun.com](#) в установочный пакет Debian. Итак, по порядку.

Установка JDK 6.0 на Debian Lenny

Есть 3 пути установки JDK в системе:

1. С сайта [java.sun.com](#) взять последнюю версию JDK (в данном случае нужен именно JDK, что бы Sun у себя на сайте не утверждал). В моем случае это `jdk-6u17-linux-x64.bin`. Запустить установку, потом все ссылки и переменные прописать в системе вручную:

```
$ sudo sh ./jdk-6u17-linux-x64.bin
```

Явно не "Debian way", на серьезной системе делать такое не хочется категорически.

2. Установить JDK через **apt-get**:

```
$ sudo apt-get update
$ sudo apt-get install sun-java6-jdk
$ sudo update-java-alternatives \-s java-6-sun
```

На десктопе это вполне приемлемое решение, но когда я увидел что установщик хочет притащить на сервер...

The following NEW packages will be installed:

avahi-daemon dbus dbus-x11 defoma gsfonsts gsfonsts-x11 java-common libavahi-common-data libavahi-common3 libavahi-core5 libdaemon0 libdbus-1-3 libfontenc1 libice6

libnss-mdns libsm6 libxfont1 libxi6 libxt6 libxtst6 locales odbcinst1debian1 sun-java6-bin sun-java6-jdk sun-java6-jre unixodbc xfonts-encodings xfonts-utils
0 upgraded, 28 newly installed, 0 to remove and 0 not upgraded.

Need to get 56.5MB/61.1MB of archives.

After this operation, 175MB of additional disk space will be used.

Do you want to continue (Yes/no)?

...при том, что о всяких "умных" вещах на сервере остались нехорошие воспоминания, я отказался.

3. Подготовить установочный пакет Debian и установить его в систему:

```
$ sudo apt-get install java-package
$ fakeroot make-jpkg jdk-6u17-linux-x64.bin
$ sudo dpkg \-i sun-j2sdk1.6_1.6.0+update1_amd64.deb
```

Именно так, **fakeroot** необходимо запускать с правами простого пользователя, по-другому не запустится. И вот тут начинается самое интересное – `make-jpkg` вылетает с ошибкой:

Detected Debian build architecture: amd64

Detected Debian GNU type: x86_64-linux-gnu

.

No matching plugin was found.

Removing temporary directory: done

Пакет не создается. Перепробовав несколько вариантов, нашел проблему. Оказывается, все корректно работает, если номер обновления состоит из 1 цифры, если же из 2 как в моем случае (17), неправильно распознается имя файла. Для лечения надо отредактировать `/usr/share/java-package/sun-j2sdk.sh` :

```
$ sudo nano /usr/share/java-package/sun-j2sdk.sh
```

И заменить (в случае JDK):

```
"jdk-6u"[0-9]"-linux-x64.bin") # SUPPORTED
j2se_version=1.6.0+update${archive_name:6:1}${revision}
"jdk-6u"[0-9][0-9]"-linux-x64.bin") # SUPPORTED
j2se_version=1.6.0+update${archive_name:6:1}${revision}
```

После этого все работает как часы, пакет создается, устанавливается и проверяется:

```
$ java -version
java version "1.6.0_17"
Java(TM) SE Runtime Environment (build 1.6.0_17-b04)
Java HotSpot(TM) 64-Bit Server VM (build 14.3-b01, mixed mode)
```

Установка JIRA

1. Создать пользователя для запуска JIRA (запускать такую систему от root'a – не очень хорошая идея):

```
$ sudo useradd \--create-home \--home-dir /home/jira \--shell /bin/bash jira
```

2. Скачать пакет установки (месячный триал), распаковать:

```
$ cd /home/jira
$ wget [http://www.atlassian.com/software/jira/downloads/binary/atlassian-jira-enterprise-4.0-standalone.tar.gz]
$ tar \-zxvf atlassian-jira-enterprise-4.0-standalone.tar.gz
```

3. Указать переменную JAVA_HOME (глобально):

```
$ sudo sh \-c "echo 'export JAVA_HOME=/usr/lib/j2sdk1.6-sun/' >> /etc/profile"
```

Если собираемся запускать от текущего пользователя, достаточно экспортировать переменную на время текущей сессии, например:

```
$ export JAVA_HOME=/usr/lib/j2sdk1.6-sun/
```

Укажите каталог Java в своей системе. Изменения в /etc/profile вступят в силу после повторного входа в систему (выход, вход).

4. Указать домашний каталог JIRA (Jira Home):

```
$ nano ./atlassian-jira/WEB-INF/classes/jira-application.properties
```

1. The jira.home configuration must be set and specifies the directory in which JIRA will store its data files.
2. This must be set to an absolute path. Relative paths are not allowed.
3. Ensure that only one JIRA instance uses the selected JIRA Home.

#####
4. Note for Windows Users

#
5. Each backslash in your path must be written as a forward slash.
6. - For example:
7. c:\jira\data
#
8. should be written as:
#
9. c:/jira/data
jira.home = /home/jira

5. Запускаем.

Если у вас в системе менее 256MB свободной памяти, даже и не пытайтесь. На системах с менее чем 1GB RAM тоже лучше не запускать.

```
$ sudo \-u jira bin/startup.sh
```

Заходим по адресу <http://yourserver:8080/>, выбираем язык (русский в наличии), вводим лицензию, настраиваем и работаем. Если что-то не работает, лог здесь: logs/catalina.out.

Оригинал статьи: <http://rat.admin.lv/?p=328>

Так же полезная статья про установку JIRA на Debian: http://kkv.spb.su/doku.php?id=howto:jira_setting_up

И еще: <http://adw0rd.ru/2012/debian-jira-and-greenhopper/>