

Установка

Все компоненты NETAMS можно запускать на одной машине: база данных mysql, биллинг, вебсервер интерфейса, сервер отчетов. В случае большой абонентской базы (более 1000 пользователей), или существенной нагрузки по трафику, возможно без особых усилий разнести эти процессы на индивидуальные сервера.

NETAMS 4.0 отлично, без изменений и каких-либо "компиляций" работает и под Windows, и под Linux/FreeBSD. Никаких компиляторов, пакетов и прочего обычно не требуется (однако см. документацию по psap). За исключением, конечно, исполняющей виртуальной машины Java (JRE, но пойдет и JDK).

Её необходимо скачать и поставить отсюда: <http://www.oracle.com/technetwork/java/index.html>. В данный момент мы работаем на Oracle JRE и OpenJDK версий 6.x (1.6.0.xx). Вы можете использовать любую другую версию JRE, но работоспособность такой конструкции не тестировалась, и не гарантируется.

Для определенности предположим, что для Windows версии вы поставили Java JDK в каталог "C:\Program Files\Java\jdk1.6.0_18", а для Linux - в "/usr/local/jre1.6.0_27". В подкаталоге "bin/java" находится собственно исполняемый файл виртуальной машины. Вы можете запустить его и проверить, какова версия:

```
C:\Program Files\Java\jdk1.6.0_18\bin>java -version
java version "1.6.0_18"
Java(TM) SE Runtime Environment (build 1.6.0_18-b04)
Java HotSpot(TM) Client VM (build 14.3-b01, mixed mode, sharing)
```

Если вы используете сбор данных (data source) через локальный механизм PCAP, почитайте [ЭТОТ ДОКУМЕНТ](#), и поставьте требуемые библиотеки.

Далее, вам необходимо скачать и распаковать дистрибутив NETAMS отсюда. Мы собираем несколько почти одинаковых дистрибутивов, с библиотеками под конкретную архитектуру и операционную систему.

Вопрос: Почему в качестве веб-контейнера вы не применяете Апач? Или нет, я люблю Tomcat или Resin.

Ответ: Апач не умеет исполнять программы на Java, это не сервлет контейнер. Вы можете использовать что угодно вместо Jetty, но настройку вам придется провести самостоятельно.

Скачанный архив необходимо распаковать куда-нибудь, например в C:\netams4 (Windows) или /usr/local/netams4 (Linux/FreeBSD). Там вы увидите список файлов, похожий на:

```
mylinux:/usr/local/netams4/# ls -al
total 2142
drwxr-xr-x 6 root root    4096 2010-07-12 20:27 .
drwxr-xr-x 4 root root    4096 2010-07-12 19:35 ..
drwxr-xr-x 2 root root    4096 2010-07-12 19:10 doc
drwxr-xr-x 5 root root    4096 2010-07-12 19:29 jserver
drwxr-xr-x 5 root root    4096 2010-07-12 19:29 repserver
drwxr-xr-x 2 root root    4096 2010-07-12 19:10 lib
-rwxr-r-  1 root root     984 2010-07-12 19:25 netams4.properties
-rwxr-xr-x 1 root root     283 2011-07-02 12:52 startup.sh
drwxr-xr-x 5 root root    4096 2010-07-12 19:35 webadmin
```

В каталоге doc - некоторые примеры, в jserver - код биллинга, в lib - общие для биллинга и вебсервера библиотеки, в webadmin - интерфейсная часть, repserver - служба генерации отчетов.

Внимание! Категорически, КАТЕГОРИЧЕСКИ запрещено использовать для распаковки дистрибутива миднайт командер (mc). Эта программа обрезает длинные имена файлов в архиве дистрибутива, из-за чего служба веб-администрирования потом просто не работает. Используйте ТОЛЬКО утилиту tar.

Теперь надо привязать Java к этому каталогу:

Для Linux:

Делаем ссылку от того места, где у нас установлен JRE, к дистрибутиву:

```
ln -s /usr/local/jre1.6.0_27 java
```

Для Windows:

Правим стартап-скрипты startup.bat, jserver\jserver-startup.bat и webadmin\webadmin-startup.bat. Находим в каждом

строку

```
set JAVA_HOME=C:\Progra~1\Java\jdk1.6.0_17
```

и заменяем последние цифры, чтобы они соответствовали версии установленной у вас JRE/JDK:

```
set JAVA_HOME=C:\Progra~1\Java\jdk1.6.0_18
```

Требуется постараться, и не перепутать \jre... и \jdk... Будем надеяться, что объяснять что такое, и откуда берется "C:\Progra~1", не надо.

Дополнительно для FreeBSD:

Выполнить действия, аналогичные установке для Linux. В силу особенности реализации нативных библиотек для работы программ Java под FreeBSD, вам придется выполнить ряд дополнительных действий.

Во-первых, поддержка JParserLocalDataSource под FreeBSD реализована экспериментально. Библиотека jnetpcap (которую мы вынуждены использовать), не тестировалась для данной операционной системы. Вместо неё, лучше используйте RemoteAgentDataSource локально.

Во-вторых, необходимо убедиться, что применяемая нами библиотека libsigar может найти все зависимости - это особенно актуально для FreeBSD 8. Можно посмотреть, что нужно:

```
[root@fbsd /home]# uname -a
FreeBSD fbsd.fnm.msu.ru 8.2-RELEASE FreeBSD 8.2-RELEASE #0: Fri Feb 18
02:24:46 UTC 2011
root@almeida.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC i386
[root@fbsd /home]# ldd
/usr/home/netams4/jserver/lib/so-dll/freebsd-i386/libsigar-x86-freebsd-8
.so
/usr/home/netams4/jserver/lib/so-dll/freebsd-i386/libsigar-x86-freebsd-8
.so:
libkvm.so.3 => not found (0x0)
```

А в наличии скорее всего есть только:

```
[root@fbsd /home]# ls -la /lib/libkvm* /usr/lib/libkvm*
-r--r--r-- 1 root wheel 32780 Feb 18 04:50 /lib/libkvm.so.5
-r--r--r-- 1 root wheel 42652 Feb 18 04:50 /usr/lib/libkvm.a
lrwxr-xr-x 1 root wheel 16 Feb 18 04:50 /usr/lib/libkvm.so ->
/lib/libkvm.so.5
-r--r--r-- 1 root wheel 43552 Feb 18 04:50 /usr/lib/libkvm_p.a
```

Решаем проблему:

```
[root@fbsd /home]# ln -s /usr/lib/libkvm.so /lib/libkvm.so.3
[root@fbsd /home]# ldd
/usr/home/netams4/jserver/lib/so-dll/freebsd-i386/libsigar-x86-freebsd-8
.so
/usr/home/netams4/jserver/lib/so-dll/freebsd-i386/libsigar-x86-freebsd-8
.so:
libkvm.so.3 => /lib/libkvm.so.3 (0x281dd000)
libc.so.7 => /lib/libc.so.7 (0x28099000)
```

Этого вполне достаточно. Если так не сделать, вы при запуске jserver наверняка увидите ошибку, как ниже, и мониторинг вашей операционки работать не будет:

```
Exception in thread "alarms_events" java.lang.UnsatisfiedLinkError:
org.hyperic.sigar.Cpu.gather(Lorg/hyperic/sigar/Sigar;)V
at org.hyperic.sigar.Cpu.gather(Native Method)
at org.hyperic.sigar.Cpu.fetch(Cpu.java:30)
at org.hyperic.sigar.Sigar.getCpu(Sigar.java:320)
at org.hyperic.sigar.Sigar.getCpuPerc(Sigar.java:341)
at
com.netams.netams4.AlarmEventManager.systemCheck(AlarmEventManager.java:
108)
at com.netams.netams4.AlarmEventManager.loop(AlarmEventManager.java:99)
at com.netams.netams4.structures.EngineThread.run(EngineThread.java:41)
at java.lang.Thread.run(Thread.java:662)
```

Дополнительно для FreeBSD 9:

Дистрибутив для FreeBSD ориентирован на версию 8, как наиболее распространенную. При работе под версией 9 необходимо сделать символическую ссылку на динамическую библиотеку sigar:

```
[root@v-freebsd1 /usr/local/netams4/lib/so-dll/freebsd-x64]# ls -al
total 216
drwxr-xr-x  2 root  wheel      512 12  ° 16:20  .
drwxr-xr-x  3 root  wheel      512 12  ° 16:20  ..
-rw-r--r--  1 root  wheel    210641 12  ° 16:20  libsigar-amd64-freebsd-8.so
[root@v-freebsd1 /usr/local/netams4/lib/so-dll/freebsd-x64]# ln -s
libsigar-amd64-freebsd-8.so libsigar-amd64-freebsd-9.so
[root@v-freebsd1 /usr/local/netams4/lib/so-dll/freebsd-x64]# ls -al
total 216
drwxr-xr-x  2 root  wheel      512 16  ° 15:12  .
drwxr-xr-x  3 root  wheel      512 12  ° 16:20  ..
-rw-r--r--  1 root  wheel    210641 12  ° 16:20  libsigar-amd64-freebsd-8.so
lrwxr-xr-x  1 root  wheel       27 16  ° 15:12  libsigar-amd64-freebsd-9.so
-> libsigar-amd64-freebsd-8.so
```

Далее, необходимо скачать, поставить и настроить MySQL сервер. Как это сделать, описано в соответствующих инструкциях. По окончании установки, создайте базу данных netams4, и убедитесь что нужный пользователь (по умолчанию root без пароля) может писать в неё. **Важно:** вам необходимо вручную создать базу данных! Это делается одной командой:

```
mysqladmin create netams4
```

Вопрос: Какую версию MySQL ставить?

Ответ: Мы тестировали 5.1.10. Этому отвечает Connector/J в комплекте. Без проблем должна заработать любая из 5.x ветки.

Вопрос: А я люблю Постгрес! Он же гораздо лучше!

Ответ: Мы можем добавить поддержку любой СУБД. Обратитесь на support@netams.com.

Прежде чем производить запуск jserver-a, необходимо убедиться, что в настройках mysql указано использование TCP/IP сокета для соединений. При настройке mysql как исключительно локальная база данных, в конфигурации по умолчанию прописывается опция skip-networking. Данная опция указывает что для соединения используется не сетевой сокет, а локальный unix-сокет. Опцию skip-networking в my.cnf нужно закомментировать или удалить, иначе вы рискуете в логах увидеть ошибку с указанием что приложение не может соединиться с СУБД.

Ошибка выглядит так:

```
FATAL [main] com.netams.netams4.structures.MYSQLEngineImpl - Exiting.
Configured MySQL engine is not running.
```

Убедитесь что в системе установлен пакет ttf-dejavu.

Неустановка данного пакета грозит тем что у вас могут не отображаться графики, а при попытке посмотреть изображения, выйдет 500-ая ошибка, с указанием что не может найти шрифты.

Если вы запустили приложение и уже обнаружили, то установите пакет и перезапустите веб-интерфейс.

В завершение, вам **необходимо** убедиться в правильности настройки локальной службы DNS. Компоненты netams4 общаются между собой по TCP/IP, по умолчанию используя адрес "localhost". Из-за особенностей реализации механизма RMI в виртуальной машине Java происходит определение локального имени хоста, и попытка связи с ним по IP-адресу, полученному из данного имени. Таким образом, если команда `hostname` на вашей машине выдает некоторое имя вашего хоста, например "myserver", то прямое преобразование имени в IP-адрес должно завершаться успехом, т.е. к примеру `ping myserver` должен приводить к ответам. Если это не так, оперативное решение проблемы состоит в добавлении записи:

```
127.0.0.1          myserver
```

в файл /etc/hosts

В общем случае, система готова к работе, в условиях того, что все компоненты запускаются на одном (этом) компьютере. Возможно, вам также будет необходимо настроить агент для источника данных RemoteAgentDataSource, как описано в [соответствующем разделе](#). Теперь необходимо [посмотреть, и поправить](#) при необходимости, файл настроек netams4.properties.