

Подготовка системы, установка и обновление wisla (ручная)

Установка системы

Программно-аппаратные требования

Платформа: аппаратный сервер или виртуальная машина (с учетом будущей инфраструктуры рекомендуется не менее 30 Гбайт свободного пространства на диске, минимум 8 Гбайт оперативной памяти без GUI и 10 Гбайт оперативной памяти с GUI).

Операционная система: CentOS 7, Debian 11, Ubuntu 20.04 LTS, Astra Linux Special Edition 1.6 Smolensk, RedOS 7.3, Astra Linux Special Edition 1.7(Орел и Воронеж).

Архитектура: x86_64.

Пакеты: [deb.zip](#) | [astra.zip](#) | [centos.zip](#) | [redos.zip](#)

Настройка ОС

Ниже описаны шаги по подготовке окружения операционной системы к выполнению программы установки.

1. Установка и запуск клиента SSH

Для Linux-совместимых операционных систем можно воспользоваться стандартной консолью и утилитой ssh, авторизоваться можно как Administrator.

2. Настройка *hostname*

Задайте имя сервера в файлах `/etc/hostname` и `/etc/hosts` как указано на примерах ниже.

Пример структуры файла `/etc/hostname`:

```
wisla
```

Пример структуры файла `/etc/hosts`:

```
127.0.0.1    localhost
192.168.159.136 wisla

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
```

```
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Обратите внимание!

⚠ В некоторых Linux-дистрибутивах в файле `/etc/hosts` указанный во время установки системы `hostname` может ссылаться на `127.0.1.1`, для корректной работы сервисов **WiSLA** эту запись нужно изменить в соответствии с примером выше.

Если сетевые настройки получены по DHCP, в будущем могут возникнуть проблемы при изменении IP-адреса сервера, так как сервисы будут ссылаться на записи в файле `/etc/hosts`, который останется без изменений. Рекомендуется использовать статический IP-адрес на сетевом интерфейсе сервера.

✓ В файле `/etc/hosts` имя хоста должно соответствовать IP-адресу, отличному от `127.0.0.1` и `127.0.1.1`.

3. Создание пользователя “wisla”

Если на вашем сервере присутствует только учётная запись суперпользователя Administrator, тогда вам необходимо создать сервисную учётную запись для работы с системой. В данном примере будет создана учётная запись `wisla`:

```
sudo useradd -d /home/wisla -m wisla && sudo passwd wisla
```

В терминале сервера появится запрос на ввод пароля, задайте надёжный пароль для сервисной учётной записи.

4. Изменение привилегий для пользователя “wisla”

Чтобы все сервисы **WiSLA** работали корректно необходимо предоставить сервисной учётной записи привилегированный доступ без запроса пароля:

```
cat << EOF > /etc/sudoers.d/wisla
## Allow wisla to run any commands anywhere
wisla  ALL=(ALL:ALL) NOPASSWD:ALL

EOF
```

Данное действие обязательно, иначе могут возникнуть проблемы из-за того, что группа, в которой состоит пользователь, не имеет `NOPASSWD` и будут унаследованы её права.

5. Создайте подкаталог `/opt/wisla5`

Для хранения файлов системы **WiSLA** необходимо создать подкаталог `/opt/wisla5`:

```
# Если вы работаете из под учётной записи Administrator переключитесь на ранее созданного пользователя wisla
su -l wisla

# Если вы уже переключились на сервисную учётную запись используйте sudo
sudo mkdir -p /opt/wisla5 && sudo chown wisla:wisla /opt/wisla5
```

Скопируйте файлы дистрибутива `wisla*.run` с помощью программы **winSCP** или другим доступным способом в подкаталог `/home/wisla/`:

```
mv wisla*.run /home/wisla
sudo chown -R wisla:wisla /home/wisla/
chmod +x /home/wisla/wisla*
```

6. Установка зависимостей

В зависимости от операционной системы на вашем сервере список необходимых пакетов может отличаться, нажмите на соответствующий блок для получения информации.

RedOS

С доступом к сети или внутреннему репозиторию:

```
sudo yum install ntp lzo dialog rsync uuid zip unzip wget tar python3 fontconfig curl pv uuid python3-paramiko
```

Без доступа к сети:

```
#Копируем архив на хост удобным способом

# Разархивируем
unzip redos.zip
cd redos

# Устанавливаем
#Все по очереди:
for i in $(ls *.rpm)
do
    rpm -i $i || exit
done
#Вручную:
rpm -i libtomcrypt-1.18.2-1.el7.x86_64.rpm libtommath-1.2.0-3.el7.x86_64.rpm dialog-1.3-14.20171209.el7.x86_64.rpm
pv-1.6.6-1.x86_64.rpm uuid-1.6.2-26.el7.x86_64.rpm
rpm -i python3-pynacl-1.5.0-1.el7.x86_64.rpm python3-bcrypt-3.2.2-1.el7.x86_64.rpm python3-paramiko-3.2.0-1.el7.noarch.rpm
```

CentOS

С доступом к сети:

```
sudo yum install ntp lzo dialog rsync uuid zip unzip wget tar python3 fontconfig curl
wget https://bootstrap.pypa.io/get-pip.py
python3 get-pip.py
python3 -m pip install --upgrade pip
pip install paramiko
sudo rpm -i http://www.ivarch.com/programs/rpms/pv-1.6.6-1.x86_64.rpm
```

Без доступа к сети:

```
#Копируем архив на хост удобным способом
```

```
# Разархивируем
unzip centos.zip
cd centos-pgks

# Устанавливаем
#Все по очереди:
for i in $(ls *.rpm)
do
    rpm -i $i || exit
done

#Вручную:
rpm -i wget-1.14-18.el7_6.1.x86_64.rpm
rpm -i uuid-1.6.2-26.el7.x86_64.rpm
rpm -i rsync-3.1.2-10.el7.x86_64.rpm
rpm -i python-crypto-2.6.1-1.el7.centos.x86_64.rpm
rpm -i python-paramiko-2.1.1-9.el7.noarch.rpm
rpm -i pv-1.6.6-1.x86_64.rpm
rpm -i dialog-1.2-5.20130523.el7.x86_64.rpm

# Возможно потребуется установить дополнительные пакеты из этого архива
```

Debian и Ubuntu

```
sudo apt install -y ntp pv liblz2-2 dialog rsync uuid zip unzip wget tar python3 python3-paramiko fontconfig curl language-pack-ru
```

Astra Linux 1.6 и 1.7

С доступом к сети:

```
sudo apt install -y ntp liblz2-2 dialog rsync zip unzip wget tar python3 python3-paramiko fontconfig curl
```

Обратите внимание!



В репозиториях **Astra Linux** нет пакетов `pv`, `paramiko` и `uuid`, поэтому их необходимо установить из исходников используя `.deb`-пакеты.

Следуйте приведённой ниже инструкции в секции "**Без доступа к сети**".

Без доступа к сети:

```
unzip astra.zip
cd astra-pkgs
#Все по очереди:
ls *.deb > /tmp/packages.list && sudo dpkg -i $(cat /tmp/packages.list) && rm -rf /tmp/packages.list
#Вручную:
sudo dpkg -i pv_1.6.6-1_amd64.deb
sudo dpkg -i rsync_3.1.3-6+ci202302061937+astra1_amd64.deb
```

```
sudo dpkg -i liboss-puuid16_1.6.2-1.5+b4_amd64.deb
sudo dpkg -i uuid_1.6.2-1.5+b4_amd64.deb
sudo dpkg -i wget_1.20.1-1.1_amd64.deb
sudo dpkg -i python3-paramiko_2.6.0-1~bpo10+1_all.deb
sudo dpkg -i python3-*
sudo dpkg -i dialog_1.3-20190211-1_amd64.deb
```

Установка и настройка pip для Python

Обязательно для работы Paramiko на Astra Linux 1.7

Скачайте установочный скрипт get-pip.py (Версии могут отличаться от обновления Astra Linux, сравнивать с версией Python в системе):

```
wget https://bootstrap.pypa.io/pip/3.7/get-pip.py
```

Запустите скрипт для установки pip:

```
python3 get-pip.py
```

Установите необходимые зависимости:

```
sudo apt install python3-distutils
```

Добавьте путь к локальным бинарным файлам в переменную PATH:

```
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.bashrc
source ~/.bashrc
```

Обновите pip до последней версии:

```
python -m pip install --upgrade pip
```

Обновите библиотеку Paramiko:

```
pip install --upgrade paramiko
```

Astra Linux 1.8

С доступом к сети:

```
sudo apt install -y rsync liboss-puuid16 uuid wget python3 python3-paramiko dialog dialog fontconfig liblz2-2 python3-asn1crypto
python3-bcrypt python3-cffi-backend python3-cryptography python3-nacl rsync unzip zip
```

Обратите внимание!



В репозиториях **Astra Linux** нет пакета `pv`, поэтому его необходимо установить из исходников используя `.deb-пакет`.

Следуйте приведённой ниже инструкции в секции "**Без доступа к сети**".

Без доступа к сети:

```
unzip astra.zip
cd astra-pkgs
#Все по очереди:
ls *.deb > /tmp/packages.list && sudo dpkg -i $(cat /tmp/packages.list) && rm -rf /tmp/packages.list
#Вручную:
sudo dpkg -i pv_1.6.6-1_amd64.deb
sudo dpkg -i rsync_3.1.3-6+ci202302061937+astra1_amd64.deb
sudo dpkg -i libosp-uuid16_1.6.2-1.5+b4_amd64.deb
sudo dpkg -i uuid_1.6.2-1.5+b4_amd64.deb
sudo dpkg -i wget_1.20.1-1.1_amd64.deb
sudo dpkg -i python3-paramiko_2.6.0-1~bpo10+1_all.deb
sudo dpkg -i python3-*
sudo dpkg -i dialog_1.3-20190211-1_amd64.deb
```

Ручная установка пакета `pv`:

```
sudo dpkg -i pv_1.6.6-1_amd64.deb
```

Alt Linux (Simply Linux)

С доступом к сети:

```
sudo apt-get install -y ntp pv dialog rsync zip unzip wget tar python3 fontconfig curl python3-module-paramiko ossp-uuid
```

В ходе тестирования было замечено, что для установки **WiSLA** в **altLinux** необходимо подключаться по **ssh**, либо используйте `sudo su $(whoami)`, иначе будет возникать окно с авторизацией. Также необходимо увеличить размер `/tmp` в `/etc/fstab`, добавьте через запятую `size=4G` (**4G** указаны в качестве примера) и `mount -o remount, rw /tmp`.

7. Установить **python3** по умолчанию

Укажите системе использовать `python3` в качестве основной версии:

```
sudo update-alternatives --install /usr/bin/python python /usr/bin/python3 1
```

8. Настройка правил **firewall**

В зависимости от операционной системы на вашем сервере стандартные утилиты для управления сетевым фильтром будут отличаться, нажмите на соответствующий блок для получения информации.

CentOS

Правила для firewalld:

```
sudo firewall-cmd --permanent --zone=public --add-port=8080/tcp
sudo firewall-cmd --reload
```

Настройка SELinux:

Настройки **SELinux** по умолчанию могут блокировать доступ к серверу с системой **WiSLA**, рекомендуется настроить режим `Permissive`.

```
# Открываем файл
sudo nano /etc/selinux/config

# Устанавливаем значение и сохраняем
SELINUX=permissive

# Выключаем на текущий момент чтобы не перезагружаться
setenforce 0
```

Debian, Ubuntu и Astra Linux

Правила для UFW:

```
sudo ufw allow 8080/tcp
```

9. Настройка *limits.conf*:

Выполните команду ниже или создайте файл вручную как указано на примере ниже:

```
cat << EOF > /etc/security/limits.d/wisla
wisla soft  nfile 32768
wisla hard  nfile 32768
wisla soft  nproc 32768
wisla hard  nproc 32768

EOF
```

10. Настройка *locale*

В зависимости от операционной системы на вашем сервере набор команд может отличаться, нажмите на соответствующий блок для получения информации.

CentOS

Выполните команды приведённые ниже:

```
sudo dnf install glibc-locale-source glibc-langpack-ru
sudo localectl set-locale LANG=ru_RU.UTF-8
```

Затем заново авторизуйтесь на сервере.

Ubuntu

Выполните команды приведённые ниже:

```
sudo apt install -y locales
sudo sed -i 's|# ru_RU.UTF-8 UTF-8|ru_RU.UTF-8 UTF-8|g' /etc/locale.gen
sudo locale-gen ru_RU
sudo locale-gen ru_RU.UTF-8
sudo update-locale
localectl set-locale LANG=ru_RU.UTF-8
```

Затем заново авторизуйтесь на сервере.

Debian

Выполните команды приведённые ниже:

```
sudo locale-gen ru_RU.UTF-8
sudo dpkg-reconfigure locales
```

Затем заново авторизуйтесь на сервере.

Обратите внимание!

Перед запуском программы установки следует выполнить команду `locale` и убедиться, что активна `ru_RU.UTF-8`. При возникновении проблем необходимо обратиться к документации по дистрибутиву для установки нужной локали. Также следует проверить вывод `timedatectl`, часовой пояс должен иметь буквенное обозначение вместо `n/a`.

11. Подготовка системы к установке


Сгенерируйте SSH-ключ для беспарольного доступа по SSH для пользователя `wisla`:

```
# Переключитесь на пользователя wisla, если вы не сделали этого ранее
su -l wisla
# Сгенерируйте SSH-ключ
ssh-keygen -P ""
# В случае если установка в кластере нужно выполнить следующую команду для каждого сервера,
# где вместо $(hostname) dns имя или ip адрес в зависимости от того как будут указаны сервера в конфигурации при установке
username=$(whoami)
ssh-copy-id $username@$(hostname)
```



```
ssh-copy-id $username@localhost
# Проверьте работу аутентификации по ключам
ssh $username@$hostname exit
ssh $username@localhost exit
```

Обратите внимание!

Запроса пароля быть не должно! Если пароль запрашивается, тогда требуется найти  причину и добиться входа без пароля, иначе в процессе установки будут происходить многократные запросы пароля. Причиной может быть неразрешённый тип ключа или несоответствие сетевого (доменного) имени.

Отключите опцию ***KillUserProcesses***:

```
sudo sed -i 's/#KillUserProcesses=yes/KillUserProcesses=no/g' /etc/systemd/logind.conf
```

Перезагрузите сервер, чтобы применить изменения:

```
sudo reboot
```


Проделав указанные выше действия ваша операционная система подготовлена к запуску программы установки.


12. Запуск программы установки

Программа установки позволяет выполнить установку, настройку, обновление, удаление, запуск и остановку системы и её компонентов, резервное копирование и восстановление, а также предоставляет централизованный доступ к журналам работы. В случае распределённой или отказоустойчивой схемы установки программа запускается на одном из серверов, остальные серверы перечисляются в её настройках.

Внесение изменений в настройки работающей системы должно производиться через интерфейс программы установки. В этом случае они будут корректно внесены в соответствующие конфигурационные файлы системы и сохранены при обновлении системы.

Обратите внимание!

 Программа установки должна запускаться под сервисной учётной записью в её окружении, в данном примере это пользователь `wisla`.

В ходе тестирования выявлено, что при развёртывании окна терминала на весь экран  программа установки не запустится.

Чтобы запустить установку не разворачивайте окно на весь экран!

Если установка системы будет аварийно прервана или завершена с ошибкой, журналы установки можно найти в каталоге с программой (`install*.log`, `runtime.log`). Информация о ходе установки также доступна в буфере эмулятора терминала.

Переключитесь на каталог, в который была скопирована программа установки:

```
cd /home/wisla
```

Запустите программы установки от имени пользователя `wisla` выполнив команду ниже:

```
./wisla*.run
```



Если программа установки не стартует попробуйте выполнить `export TERM=xterm` перед её запуском.

Если приложение не запускается, следует проанализировать сообщения об ошибках и созданные в текущем каталоге log-файлы.

Навигация в программе установки осуществляется с помощью стрелок управления курсором, клавиш `Home`, `End`, `Tab`, `Esc` и `Enter`.

- ❗ Если требуется аварийно прервать работу программы, можно использовать комбинацию клавиш `CTRL+C`, для штатного завершения программы установки следует использовать кнопку `Exit`.

В процессе установки вам также необходимо проверить следующие конфигурации:

Окно 'Installer startup configuration'

Проверьте параметры '*Install master*', при установке всех компонентов на один сервер его имя должно быть указано здесь.

Окно 'Select action'

Нажмите '*Install*'.

Окно 'JRE* configuration'

Нажмите '*OK*'.

Окно 'Postgresql* configuration'

Проверить параметр '*Trust host or network*'.

Нужно проверить и заполнить *Trusted network/host*, иначе будут проблемы с подключением *Postgres* и патчами.

Окно 'Wildfly* configuration'

Проверить значение *memory size*.

Окно 'Hadoop* configuration'

Проверить имя *hostname* в '*HDFS master*' и '*Tracker host fields*'.

Окно 'HBase* configuration'

Проверить имя *hostname* в '*Zookeeper quorum*'.

Окно 'wiSLA* data collection configuration'

Если планируется использование зондов *wiProbe*, нужно прокрутить список и изменить настройку «*wiProbe destination*». В ней задаётся адрес, который будет использоваться зондом для отправки данных в систему *wiSLA*, в форме *URL*. Остальные параметры менять без необходимости не рекомендуется.

Окно 'wiSLA* LDAP configuration'

Если не планируется интегрировать систему с *MS Active Directory* или *OpenLDAP Server*, рекомендуется оставить значения по умолчанию.

Окно 'wiSLA* resources configuration'

Убедиться, что имя hostname указано в **URL**.

Окно 'wiSLA* notification and ASI configuration'

Требуется указать параметры подключения к почтовому серверу. Если этого не сделать, новые пользователи не смогут получать письма о добавлении учётной записи и другие уведомления, отсылаемые на email. Также здесь можно включить отправку SNMP-уведомлений по определённым событиям.

Настройка email-уведомлений:

1.Необходимо выбрать почту с которой буду отправляться уведомления и выполнить настройку по инструкции.

2.Устанавливаем конфиги:

- Notification enabled: true
- Profile-status notification enabled: false
- Service notification enabled: false
- Wisla notification op_link: wisla
- Wisla notification cp_link: wisla
- Wisla inter-hop master:
- Wisla inter-hop slaves:
- Wisla inter-hop slave: false
- Mail host: smtp.{домен почты}.ru
- Mail from: email (например: test_push@yandex.ru)
- Mail from alias: email без домена (например: test_push)
- Mail port: 587
- Mail protocol: smtp
- Mail smtp auth: true
- Mail smtp starttls: true
- Mail user: email с которого планируется отправка уведомлений (например: test_push@yandex.ru)
- Mail password: "пароль приложения" сформированный на шаге 1
- ASI notification enabled: false
- ASI handlers: genericSnmp
- ASI SNMP distation:
- Events limit for notification: 10
- No data duration: 10
- Reports use en filenames: false

Окно 'wiSLA* operator portal configuration'

Обращаем ваше внимание, если вы получаете доступ к порталу с помощью проброса портов или через прокси сервер, то вам необходимо отредактировать пункт **HOST** и в **Whitelisted domains** установить необходимые IP-адреса.

1. Подтверждение настроек

На этом этапе можно вернуться назад и внести исправления в настройки. После подтверждения начинается процесс установки.

Процесс установки

Во время установки в каталог `/opt/wisla5` будут добавлены следующие компоненты:

- *Zookeeper*;
- *Hadoop*;
- *HBase*;
- *PostgreSQL*;
- *Java Runtime Environment*;
- *WildFly Application Server*;
- *wiSLA Portal*.

Процесс можно прервать, нажав `CTRL+C`, все настройки будут сброшены.

После завершения установки будет предложено добавить систему в список автозагрузки – нажмите кнопку `Нет`.

Обратите внимание!

⚠ В ходе тестирования выявлено, что на *Astra Linux* в некоторых сценариях не создается `systemd unit`, при возникновении данной проблемы нужно создать его руками.

Выполните команду ниже или создайте файл вручную с содержанием как на примере ниже:

```
cat << EOF > /etc/systemd/system/wisla.service
[Unit]
Description=Starts JBoss process with wiSLA 5 system
After=network-online.target
Requires=network-online.target

[Service]
Type=forking
RemainAfterExit=true
WorkingDirectory=/opt/wisla5
ExecStart=/opt/wisla5/scripts/wisla5.sh start
ExecStop=/opt/wisla5/scripts/wisla5.sh stop

[Install]
WantedBy=multi-user.target

EOF

sudo systemctl daemon-reload
sudo systemctl enable wisla.service
```

13. Запуск WiSLA

Выйдите из программы установки и дождитесь завершения процесса в фоне.

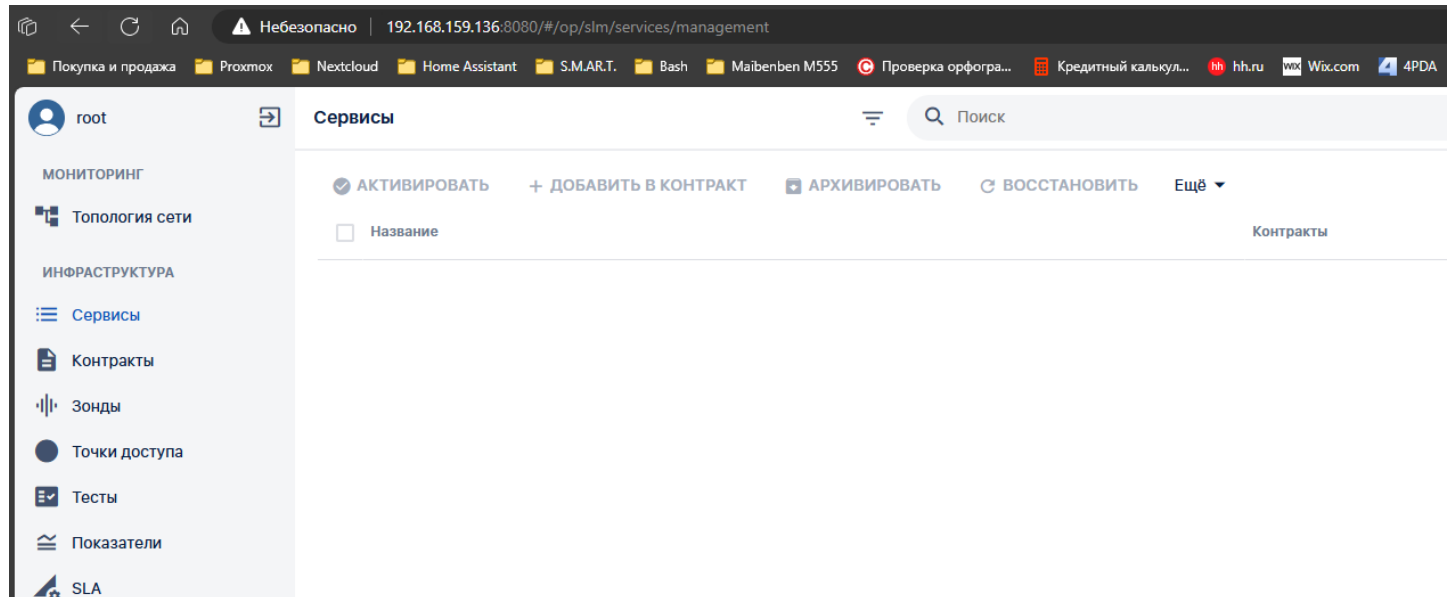
Первичный запуск системы может занимать до двух минут, ход установки можно отследить в журналах работы:

```
less -f /opt/wisla5/wildfly/current/standalone/log/server.log
less -f /opt/wisla5/wildfly/current/standalone/log/communicator.log
```

Маркером успешного запуска является следующее сообщение в журнале (server.log):

```
INFO [com.wellink.wisla.communicator.impl.state.AvailabilitySystemStateSingletonImpl] (http-0.0.0.0-0.0.0.0-8080-1) !*** THE ALL wiSLA COMPONENTS ARE FULLY DEPLOYED, INTERCONNECTED AND READY TO WORK! ***!  
13:48:30,028 INFO [com.wellink.wisla.communicator.impl.state.AvailabilitySystemStateSingletonImpl] (http-0.0.0.0-0.0.0.0-8080-1) !  
*****  
*****!
```

Теперь можно запустить веб-браузер и открыть страницу системы указав IP-адрес сервера и порт.



В данном примере система будет доступна по адресу <http://192.168.159.136:8080>.

Обновление wiSLA

Рекомендуется обновляться с предыдущей минорной версии **wiSLA** (5.1->5.2->5.2.1->5.2.2->5.2.3).

- 1) Запустить программу установки wiSLA 5.2.3;
- 2) В основном меню выбрать пункт Update;
 - 2.1) Подтвердить или отклонить создание резервной копии (рекомендуется сделать);
 - 2.2) Подтвердить остановку компонентов wiSLA;
 - 2.3) После загрузки настроек системы, в каждом окне проверить настройки (по необходимости внести правки) и подтвердить для продолжение установки;
 - 2.4) После обновления и запуска всех компонентов системы подтвердить или отклонить добавление wiSLA в автозагрузку;
- 3) После успешного запуска сервера приложений, выполнить индексацию ([Maintenance > wiSLA > Reindex](#)) (*Не путать со Standalone Reindex!*);
- 4) Открыть портал, проверить работу системы;
- 5) Очистить кэш браузера на всех рабочих местах.

Возможные ошибки в процессе обновления:

Иногда при обновлении до новой версии может зависнуть сервис `wildfly`, на это будут указывать следующие записи в журнале `server.log`:

```
13:24:21,676 ERROR [org.jboss.as.controller.management-operation] (DeploymentScanner-threads - 2)
WFLYCTL0013: Operation ("deploy") failed - address: ([("deployment" => "wisla-engine-5.2.4-SNAPSHOT.war")]) - failure description:
{"WFLYCTL0080: Failed services" => {"jboss.deployment.unit.\"wisla-engine-5.2.4-SNAPSHOT.war\".undertow-deployment" =>
"java.lang.RuntimeException: org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with name
'expireSessionSchedulerSingleton': Unsatisfied dependency expressed through field 'eventLoggerService'; nested exception is
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'eventLoggerService' defined in class path resource
[audit/conf/spring/services.xml]: Cannot resolve reference to bean 'hibernateAuditLogAppender' while setting bean property 'appenders' with
key [0]; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'hibernateAuditLogAppender' defined in class path resource [audit/conf/spring/services.xml]: Cannot resolve reference to bean
'sessionFactory' while setting bean property 'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException:
Error creating bean with name 'sessionFactory' defined in class path resource [engine/conf/spring/datasource.xml]: Invocation of init method
failed; nested exception is org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: java.lang.RuntimeException: org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with
name 'expireSessionSchedulerSingleton': Unsatisfied dependency expressed through field 'eventLoggerService'; nested exception is
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'eventLoggerService' defined in class path resource
[audit/conf/spring/services.xml]: Cannot resolve reference to bean 'hibernateAuditLogAppender' while setting bean property 'appenders' with
key [0]; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'hibernateAuditLogAppender' defined in class path resource [audit/conf/spring/services.xml]: Cannot resolve reference to bean
'sessionFactory' while setting bean property 'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException:
Error creating bean with name 'sessionFactory' defined in class path resource [engine/conf/spring/datasource.xml]: Invocation of init method
failed; nested exception is org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with name
'expireSessionSchedulerSingleton': Unsatisfied dependency expressed through field 'eventLoggerService'; nested exception is
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'eventLoggerService' defined in class path resource
[audit/conf/spring/services.xml]: Cannot resolve reference to bean 'hibernateAuditLogAppender' while setting bean property 'appenders' with
key [0]; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'hibernateAuditLogAppender' defined in class path resource [audit/conf/spring/services.xml]: Cannot resolve reference to bean
'sessionFactory' while setting bean property 'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException:
Error creating bean with name 'sessionFactory' defined in class path resource [engine/conf/spring/datasource.xml]: Invocation of init method
failed; nested exception is org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'eventLoggerService' defined in class
path resource[audit/conf/spring/services.xml]: Cannot resolve reference to bean 'hibernateAuditLogAppender' while setting bean property
'appenders' with key [0]; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'hibernateAuditLogAppender' defined in class path resource [audit/conf/spring/services.xml]: Cannot resolve reference to bean
'sessionFactory' while setting bean property 'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException:
Error creating bean with name 'sessionFactory' defined in class path resource [engine/conf/spring/datasource.xml]: Invocation of init method
failed; nested exception is org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'hibernateAuditLogAppender' defined
in class path resource [audit/conf/spring/services.xml]: Cannot resolve reference to bean 'sessionFactory' while setting bean property
'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name
'sessionFactory' defined in class path resource [engine/conf/spring/datasource.xml]: Invocation of init method failed; nested exception is
org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: org.hibernate.search.exception.SearchException: HSEARCH000103: Unable to initialize IndexManager named 'sap'
Caused by: org.hibernate.search.exception.SearchException: Unable to open Lucene IndexReader for IndexManager sap
Caused by: org.apache.lucene.index.CorruptIndexException: file mismatch, expected id=42hlomvwa71vvwn7vem94t0p,
got=7mz5ojw75crmjs1kxhmc322p2 (resource=BufferedChecksumIndexInput(MMapIndexInput(path=\"/opt/wisla5/wildfly/wildfly-14.0.1.Final/
bin/searchindexes/engine/sap/_q.si\")))}}}
```

Чтобы решить эту проблему удалите содержимое каталога `/opt/wisla5/wildfly/wildfly-14.0.1.Final/bin/searchindexes/engine/sap/`:

```
sudo rm -rf /opt/wisla5/wildfly/wildfly-14.0.1.Final/bin/searchindexes/engine/sap/*
```

Затем повторно запустите процесс обновления с помощью инсталлятора `wisla`, после обновления системы не забудьте запустить индексацию.

Если перед началом обновления требуется удалить данные из базы данных:

Иногда при обновлении wiSLA структура таблиц в БД может кардинально измениться и для корректной работы мониторинга потребуется повторная постановка инфраструктуры на мониторинг.

В данной ситуации нужно проделать следующие действия:

1. Сделать резервную копию БД для возможности восстановления данных на другой машине со старой версией приложения;
 2. Остановите приложения wiSLA;
 3. Подключиться к БД используя клиент, например DBviewer;
 4. Переключиться на БД wisla и выполнить скрипт `wisla_init_schema.sql`;
 5. Затем выполнить скрипт `wisla_init_schema.sql`;
 6. После выполнения скриптов отключитесь от БД и вернитесь к терминалу сервера с инсталлером;
 7. Загрузите на сервер пустую БД wiSLA `clear1.backup`;
 8. Перейдите в раздел backup и загрузите пустую БД в базу в режиме восстановления;
 9. Загрузив БД перейдите к настройке PostgreSQL (Maintenance > PostgreSQL) и запустите патч для создания необходимых таблиц;
 10. После того как таблицы будут сформированы запустите приложения wiSLA и проверьте работу веб-портала.
-