

Подготовительные этапы к установке кластера

Преимущества кластера

Настройка отказоустойчивого кластера wiSLA позволяет решить 2 задачи:

- в случае отказа одного из ЦОД система сохраняет работоспособность;
- в кластере работает балансировка нагрузки, что позволяет более эффективно использовать аппаратные ресурсы серверов.

В примере будет показана установка системы wiSLA на отказоустойчивый контур, который включает в себя семь серверов, распределённых между двумя ЦОД (по три в каждом) и одним дополнительным сервером – «третьей точкой опоры» (см. рисунок 66). Для взаимодействия между серверами выделена подсеть «межсерверного взаимодействия».

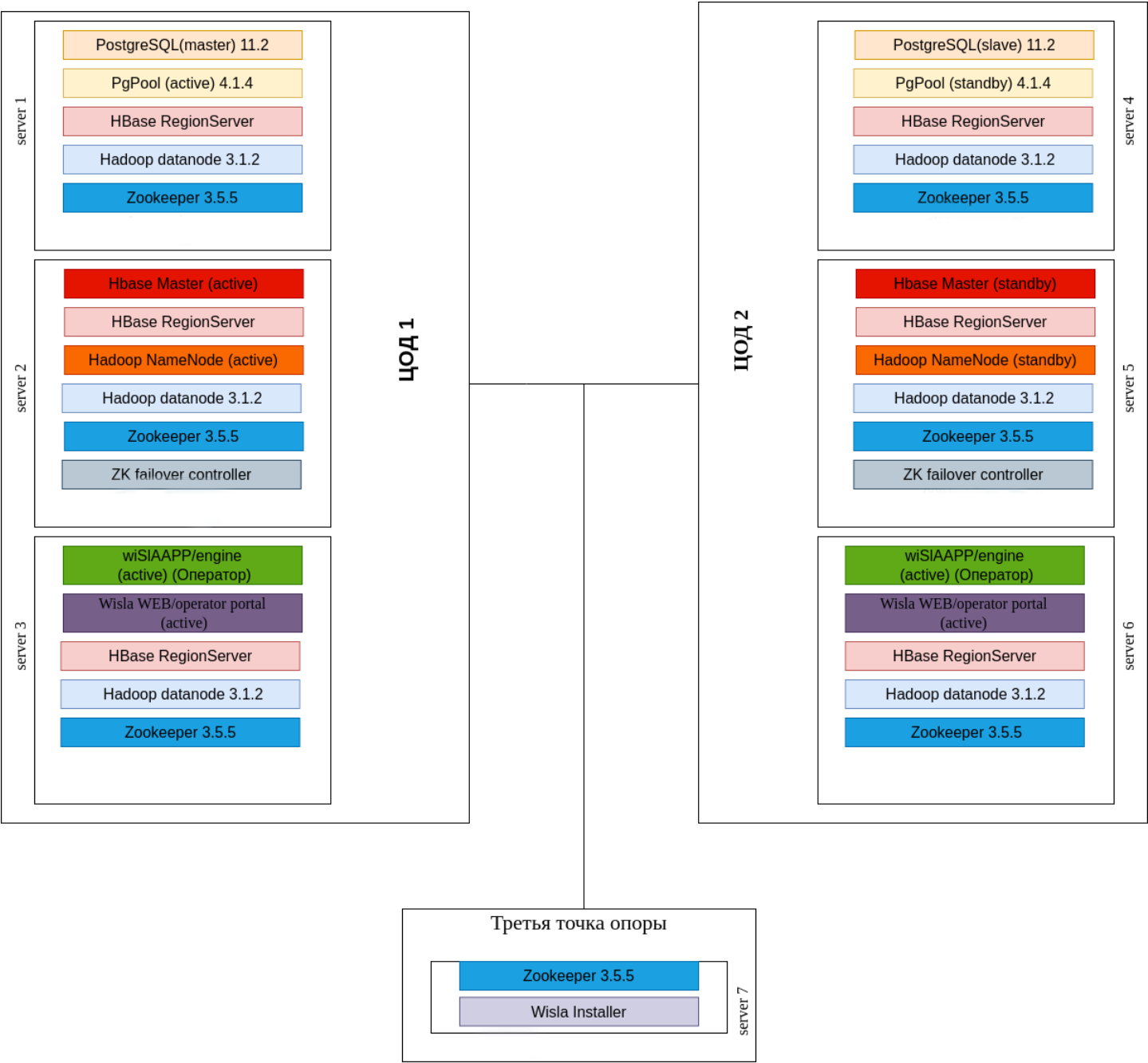


Рис. 66 Логическая группировка серверов в отказоустойчивом контуре wiSLA

Дополнительно могут быть развернуты wisla contactor portals (но не на 3 и 6 ноде, где уже есть портал оператора)

Отказоустойчивый контур должен состоять минимум из двух блоков (ЦОД), которые включают в себя сервера с основными компонентами, и дополнительного сервера (третьей точки опоры), с помощью которого контролируется доступность основных ЦОД и целостность кластера. Всего для отказоустойчивого контура должно быть выделено не менее пяти отдельных серверов. Желательно, чтобы аппаратная конфигурация серверов была одинаковой. В этом случае можно производить установку и обновление системы с помощью программы установки без ручного изменения параметров распределения оперативной памяти по компонентам.

В таблице приведён пример настройки отказоустойчивого контура, который содержит два ЦОД (по три сервера на каждом) и один дополнительный сервер (третья точки опоры).

Таблица 8 – Топология кластера (пример).

ЦОД / сервер	Имя сервера (hostname) / IP-адрес	Компоненты
--------------	-----------------------------------	------------

ЦОД1	wislaserver01 / 192.168.1.101	PostgreSQL: Master Pgpool HBase: HRegionServer Hadoop: DataNode Zookeeper
	wislaserver02 / 192.168.1.102	HBase: HMaster, HRegionServer Hadoop: NameNode, DataNode Zookeeper
	wislaserver03 / 192.168.1.103	APP-server WEB-server HBase: HRegionServer Hadoop: DataNode Zookeeper
ЦОД2	wislaserver04 / 192.168.1.104	PostgreSQL: Slave Pgpool HBase: HRegionServer Hadoop: DataNode Zookeeper
	wislaserver05 / 192.168.1.105	HBase: HMaster, HRegionServer Hadoop: NameNode, DataNode Zookeeper
	wislaserver06 / 192.168.1.106	APP-server WEB-server HBase: HRegionServer Hadoop: DataNode Zookeeper

Дополнительный сервер	wislaserver07 / 192.168.1.107	Zookeeper, wiSLA-Installer
-----------------------	-------------------------------	----------------------------


Предварительно для каждого сервера контура требуется базовая настройка, которая описана в разделе «Подготовка операционной системы к запуску программы установки». Для серверов с Hadoop NameNode (wislaserver02 и wislaserver05) потребуется дополнительно примонтировать блочное устройство, объем диска на обоих серверах должен быть одинаковым. Для серверов с Prgpool (wislaserver01 и wislaserver04) должен быть выделен IP-адрес в подсети межсерверного взаимодействия, который будет использовать Prgpool. Далее в примерах настройки используется «192.168.1.110».


На серверах с Prgpool (wislaserver01 и wislaserver04) также требуется убедиться в отсутствии alias-интерфейса

«eth0:10», где eth0 – корневой интерфейс, на котором настроена подсеть межсерверного взаимодействия, фактически может отличаться.

Настройка беспарольного доступа по SSH для межсерверного взаимодействия

Для корректной установки системы wiSLA и взаимодействия серверов контура должен быть организован беспарольный доступ по SSH под пользователем wisla по принципу «каждый с каждым». Также требуется беспарольный доступ под пользователем root между серверами с Prgpool и третьей точкой опоры. При этом для доступа с сервера на сервер используется hostname. Для этого требуется выполнить следующие шаги:

 **Внимание, возможно требуется после копирования длинных команд - заменять пробелы на пробелы**

 **Внимание, при изменении топологии, возможно корректирование предлагаемых команд**

1. На 7-й сервере добавить хосты в /etc/hosts

```
mkdir installDir; pushd installDir
host_prefix=0001wislatest0
echo 192.168.1.101 "$host_prefix"1 >> hostsForALL
echo 192.168.1.102 "$host_prefix"2>> hostsForALL
echo 192.168.1.103 "$host_prefix"3>> hostsForALL
echo 192.168.1.104 "$host_prefix"4>> hostsForALL
echo 192.168.1.105 "$host_prefix"5>> hostsForALL
echo 192.168.1.106 "$host_prefix"6>> hostsForALL
echo 192.168.1.107 "$host_prefix"7>> hostsForALL
cat hostsForALL | sudo tee -a /etc/hosts
hostnames="0001wislatest01 0001wislatest02 0001wislatest03 0001wislatest04 0001wislatest05 0001wislatest06 0001wislatest07"
hostnames_1_6="0001wislatest01 0001wislatest02 0001wislatest03 0001wislatest04 0001wislatest05 0001wislatest06"
hostnames_1_4_7="0001wislatest01 0001wislatest04 0001wislatest07"
```

2. На каждом сервере контура под пользователем wisla выполнить генерацию ключей, пароль оставить пустым (можно вообще 1 ключ создать и скопировать его везде):

```
su -l wisla
cd /home/wisla
ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa # или ssh-keygen -t rsa
```

Если не известен пароль, то необходимо:

```
зайти на хост на который нужно подключиться
скопировать содержимое ~/.ssh/*.pub #ssh-keygen -t rsa если нет файла
добавить в конец файла ~/.ssh/authorized_keys
```

4. Скопировать /etc/hosts на каждый хост

```
for i in $hostnames_1_6; do cat hostsForALL | ssh "$i" 'sudo tee -a /etc/hosts';done
for i in $hostnames_1_6; do ssh "$i" "sudo hostnamectl set-hostname $i";done
```

5. С каждого сервера контура под пользователем wisla выполнить копирование ключа на все остальные сервера для пользователя wisla и для пользователя root для части серверов (На серверах с Pgrpool и дополнительном сервере под пользователем wisla выполнить копирование ключа на другие сервера с Pgrpool или дополнительный сервер для пользователя root):

```
host_keys="$(for i in $hostnames;do ssh \"$i\" 'cat /home/wisla/.ssh/*.pub'; done)"

for i in $hostnames;do echo "$host_keys" | ssh "$i" tee -a /home/wisla/.ssh/authorized_keys; done

host_keys="$(for i in $hostnames_1_4_7;do ssh "$i" 'cat /home/wisla/.ssh/*.pub'; done)"

for i in $hostnames_1_4_7;do echo "$host_keys"| ssh "$i" 'sudo mkdir -p /root/.ssh; sudo tee -a /root/.ssh/authorized_keys'; done
```

6. Выполнить разовое подключение по SSH под пользователем wisla с каждого сервера на другие сервера, чтобы подтвердить добавление ключей удалённых серверов (по умолчанию отпечаток ключа будет храниться в файле /home/wisla/.ssh/known_hosts). При этом на вопрос «Are you sure you want to continue connecting (yes/no)?» требуется отвечать «yes». При подключении пароль удалённого сервера запрашиваться не должен.

Выполнить команду на каждом сервере под пользователем wisla:

```
for i in $hostnames; do ssh "$i" exit; done
```

Выполнить команду на 1 4 7 серверах под пользователем wisla:

```
for i in $hostnames_1_4_7; do ssh root@"$i" exit; done
```

В некоторых случаях доступ по root из ssh невозможен: следует проверить файл /etc/security/access.conf(ROOT:ALL) и файл /etc/ssh/sshd_config (PermitRootLogin yes)

Установка GlusterFS

GlusterFS – распределённая, параллельная, линейно масштабируемая файловая система с возможностью защиты от сбоев. Предварительно на серверах wislaserver02 и wislaserver05 должны быть созданы lvm-разделы абсолютно одинакового размера. Далее в примере используется «/dev/sdb1», фактически название раздела может отличаться.

Действия по установке и настройке GlusterFS выполняются под пользователем root.

Для установки GlusterFS на серверах wislaserver02 и wislaserver05 требуется выполнить следующие шаги:

1. Загрузить установочные пакеты GlusterFS по ссылке <ftp://ftp.wellink.ru/Deploies/Centos/cluster/glusterfs.tar.gz> ;
2. Подключиться к серверу по SSH под пользователем root;

3. Перейти в каталог с архивом, распаковать архив, установить необходимые пакеты glusterfs, glusterfs-server, glusterfs-cli,:

```
tar zxvf glusterfs.tar.gz
yum install glusterfs-3.5.2-1.el6.x86_64.rpm glusterfs-api-3.5.2-1.el6.x86_64.rpm
glusterfs-cli-3.5.2-1.el6.x86_64.rpm glusterfs-fuse-3.5.2-1.el6.x86_64.rpm glusterfs-georeplication-3.5.2-1.el6.x86_64.rpm glusterfs-
libs-3.5.2-1.el6.x86_64.rpm glusterfsserver-3.5.2-1.el6.x86_64.rpm
```

Настройка GlusterFS

Для настройки работы GlusterFS требуется выполнить следующие шаги:

1. Рекомендовано на серверах wislaserver02 и wislaserver05 отформатировать раздел для glusterfs в файловую систему xfs:

```
mkfs.xfs /dev/sdb1
```

2. Смонтировать раздел, добавить в автозапуск и запустить glusterfs на wislaserver02 и wislaserver05:

```
mkdir -p /mnt/gluster && mount /dev/sdb1 /mnt/gluster && mkdir -p /mnt/gluster/brick
echo "/dev/sdb1 /mnt/gluster xfs defaults 0 0" >> /etc/fstab
systemctl start glusterd # /etc/init.d/glusterd start
```

3. На сервере wislaserver02 выполнить команды:

```
gluster peer probe "$host_prefix"5
gluster volume create namenodevol rep 2 transport tcp "$host_prefix"2:/mnt/gluster/brick
"$host_prefix"5:/mnt/gluster/brick
gluster volume start namenodevol
mkdir /mnt/glusterVol; mount -t glusterfs "$host_prefix"2:/namenodevol /mnt/glusterVol/
chown wisla:wisla -R /mnt/glusterVol
```

4. На сервере wislaserver05 выполнить команды:

```
mkdir /mnt/glusterVol; mount -t glusterfs "$host_prefix"5:/namenodevol /mnt/glusterVol/
chown wisla:wisla -R /mnt/glusterVol
```

Внимание, следует протестировать что файл созданный на одном узле в /mnt/glusterVol ,должен появиться в том же месте на другом узле

5. На сервере wislaserver02 выполнить команды:

```
touch /mnt/glusterVol/test_file.txt
```

6. На сервере wislaserver05 проверить наличие файла и затем файл можно удалить

```
ls /mnt/glusterVol/test_file.txt
rm -f /mnt/glusterVol/test_file.txt
```