Arenadata™ Hadoop

Версия - v1.4.1

Настройка безопасности и авторизации в Hadoop

2021 Arenadata™

Оглавление

1 Функции безопасности Hadoop **3**
1.1 Безопасность периметра . . . 3
1.2 Аутентификация . . . 3
1.3 Авторизация (контроль доступа) . . . 3
1.4 Отчетность (аудит и мониторинг безопасности) . . . 4
1.5 Защита данных . . . 4

2 Настройка авторизации в Hadoop **5**
2.1 Ranger. Введение . . . 5
2.2 Установка Ranger . . . 12
2.3 HDFS Policy . . . 54

В документе приведены настройки функций безопасности Hadoop, авторизация в кластере, установка Ranger и создание HDFS Policy.

Документ может быть полезен администраторам, программистам, разработчикам и сотрудникам подразделений информационных технологий, осуществляющих внедрение кластера.

Important: Контактная информация службы поддержки – e-mail: info@arenadata.io

Глава 1

Функции безопасности Hadoop

Для организаций, которые хранят конфиденциальные данные в экосистеме **Hadoop**, например, запатентованные или персональные данные, которые подлежат соблюдению нормативных требований (**HIPPA**, **PCI**, **DSS**, **FISAM** и т.д.), безопасность очень важна. Многие организации также должны придерживаться строгой внутренней политики безопасности.

Arenadata Hadoop (далее – **ADH**) обеспечивает комплексный подход к безопасности в следующих ключевых областях: безопасность периметра, аутентификация, авторизация, отчетность, защита данных.

1.1 Безопасность периметра

ADH позволяет изолировать кластер **Hadoop** с помощью шлюза и правильно настроенных правил брандмауэра. **ADH** поддерживает следующую защиту периметра:

- Apache Knox Gateway;
- Клиенты шлюза.

1.2 Аутентификация

ADH предоставляет единую точку аутентификации для сервисов и пользователей, которая интегрируется с существующими системами идентификации предприятия и доступа. **ADH** поддерживает следующие службы аутентификации:

- Kerberos;
- LDAP;
- Локальная система Unix;
- SSO (по периметру через Apache Knox Gateway).

1.3 Авторизация (контроль доступа)

ADH предоставляет функции, позволяющие системным администраторам контролировать доступ к данным **Hadoop** с использованием авторизации на основе ролей. **ADH** поддерживает следующие модели авторизации:

- Высококачественное управление доступом для данных, хранящихся в HDFS;
- Контроль доступа на уровне ресурсов для YARN;

- Контроль доступа на уровне крупнозернистого уровня для операций MapReduce;
- Контроль доступа к семейству на уровне таблицы и столбца для данных HBase;
- Контроль доступа на уровне таблицы для наборов данных Apache Hive.

1.4 Отчетность (аудит и мониторинг безопасности)

ADH позволяет отслеживать активность **Hadoop**, используя собственный аудит (журналы аудита), журналы аудита безопасности периметра на шлюзе **Knox** и центральное расположение консоли администрирования **ADH**, включая:

- Запросы доступа;
- Операции обработки данных;
- Изменение данных.

1.5 Защита данных

ADH предоставляет механизмы для шифрования данных при их передаче и требует использования партнерских решений для шифрования данных в состоянии покоя, их поиска и маскирования. **ADH** поддерживает следующие методы шифрования:

- SSL для компонентов ADH;
- Шифрование RPC;
- Протокол передачи данных.

Глава 2

Настройка авторизации в Hadoop

- *Ranger. Введение*
- *Предварительные требования к установке*
- *Установка Ranger*
- *HDFS Policy*

2.1 Ranger. Введение

Apache Ranger можно установить при помощи пользовательского интерфейса **Ambari** или вручную, используя платформу **Arenadata Hadoop**. В отличие от ручного процесса установки, требующего выполнения ряда шагов, установка **Ranger** с использованием интерфейса **Ambari** проще и легче. Опция службы **Ranger** доступна через мастер **Add Service** после инсталляции кластера **ADH** с помощью установщика.

После установки и настройки **Ambari** можно использовать мастер добавления служб для установки следующих компонентов:

- Ranger Admin
- Ranger UserSync
- Ranger Key Management Service

После установки и запуска этих компонентов можно включить плагины **Ranger**, перейдя к каждому отдельному сервису **Ranger** (**HDFS**, **HBase**, **Hiveserver2**, **Storm**, **Knox**, **YARN** и **Kafka**) и изменив конфигурацию в расширенном режиме *ranger-<service>-plugin-properties*.

Important: При включении плагина Ranger необходимо перезапустить компонент

Important: Включение Apache Storm или Apache Kafka требует включения Kerberos

При обновлении **ADH** (на **Ambari 2.5.0**) установка **Ranger DB** выполняется при первом запуске **Ranger** (в предыдущих версиях настройка **Ranger DB** выполнялась во время установки). Это означает, что **Ranger** при первом запуске может занять больше времени (последующие перезагрузки будут такими же быстрыми, как и раньше).

2.1.1 Предварительные требования к установке

Перед установкой **Ranger** необходимо убедиться, что кластер отвечает следующим требованиям:

- Рекомендуется хранить аудиты как в HDFS, так и в Solr. Конфигурация по умолчанию для Ranger Audits в Solr использует общий экземпляр Solr, предоставляемый сервисом Ambari Infra (дополнительные сведения см. в разделе Ranger Audit Settings);
- Чтобы обеспечить принудительную авторизацию на уровне групп LDAP/AD в Hadoop, необходимо настроить сопоставление групп Hadoop для LDAP/AD для LDAP (*Настройка сопоставления групп Hadoop для LDAP/AD*);
- Должен быть запущен и доступен экземпляр базы данных MySQL, Oracle, PostgreSQL или Amazon RDS, который будет использоваться Ranger. Установщик Ranger создаст двух новых пользователей (имена по умолчанию: *rangeradmin* и *rangerlogger*) и две новые базы данных (имена по умолчанию: *ranger* и *ranger_audit*).

  Конфигурация экземпляра для Ranger для некоторых баз данных описана в следующих разделах:

  – *Конфигурация MySQL*

  – *Конфигурация PostgreSQL*

  – *Конфигурация Oracle*

  При использовании Amazon RDS есть дополнительные требования (*Требования к Amazon RDS*).
- При решении не предоставлять данные учетной записи администратора базы данных (DBA) установщику Ambari Ranger, можно использовать Python-скрипт *dba_script.py* для создания пользователей базы данных Ranger DB без предоставления этой информации установщику. После чего запустить обычную установку Ambari Ranger без указания имени и пароля администратора. Дополнительные сведения приведены в разделе **'Настройка пользователей базы данных без совместного использования учетных данных DBA'_**.

Настройка сопоставления групп Hadoop для LDAP/AD

Для обеспечения принудительной авторизации на уровне групп **LDAP/AD** в **Hadoop**, необходимо настроить сопоставление групп **Hadoop** для **LDAP/AD**.

Important: Доступ к LDAP и сведения о подключении: настройки LDAP могут различаться в зависимости от используемой реализации LDAP

Существует три способа настройки сопоставления групп **Hadoop**.

- **Настройка сопоставления групп Hadoop для LDAP/AD с использованием SSSD (рекомендуется)**

Для сопоставления групп рекомендуется использовать SSSD или один из следующих сервисов подключения ОС **Linux** к **LDAP**:

- Centrify
- NSLCD
- Winbind
- SAMBA

Большинство перечисленных сервисов позволяет не только искать пользователя и перечислять группы, но также выполнять другие действия на хосте. При этом ни одно из этих действий не требуется для сопоставления групп **LDAP** в **Hadoop**. Поэтому, оценивая эти сервисы, необходимо понимать разницу между модулем **NSS** (который выполняет разрешение пользователь/группа) и модулем **PAM** (который выполняет аутентификацию

пользователя). Для возможности поиска (или “валидации”) пользователя в **LDAP** и перечисления групп требуется **NSS**. А **PAM** может представлять угрозу безопасности.

- **Настройка сопоставления групп Hadoop в файле core-site.xml**

Настройка **Hadoop** для использования сопоставления групп на основе **LDAP** в файле *core-site.xml* осуществляется в следующем порядке:

1. Добавить свойства, показанные в приведенном ниже примере, в файл *core-site.xml*. Необходимо указать значение для привязанного пользователя, его пароль и другие свойства, специфичные для экземпляра LDAP, и убедиться, что фильтры классов объектов, пользователей и групп соответствуют значениям, указанным в экземпляре LDAP.

```
<property>
<name>hadoop.security.group.mapping</name>
<value>org.apache.hadoop.security.LdapGroupsMapping</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.bind.user</name>
<value>cn=Manager,dc=hadoop,dc=apache,dc=org</value>
</property>

<!-
<property>
<name>hadoop.security.group.mapping.ldap.bind.password.file</name>
<value>/etc/hadoop/conf/ldap-conn-pass.txt</value>
</property>
->

<property>
<name>hadoop.security.group.mapping.ldap.bind.password</name>
<value>hadoop</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.url</name>
<value>ldap://localhost:389/dc=hadoop,dc=apache,dc=org</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.url</name>
<value>ldap://localhost:389/dc=hadoop,dc=apache,dc=org</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.base</name>
<value></value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.search.filter.user</name>
<value>(&amp;(|(objectclass=person)(objectclass=applicationProcess))(cn={0}))</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.search.filter.group</name>
<value>(objectclass=groupOfNames)</value>
</property>
```

```
<property>
<name>hadoop.security.group.mapping.ldap.search.attr.member</name>
<value>member</value>
</property>

<property>
<name>hadoop.security.group.mapping.ldap.search.attr.group.name</name>
<value>cn</value>
</property>
```

2. В зависимости от конфигурации можно обновлять сопоставления пользователей и групп с помощью следующих команд HDFS и YARN:

```
hdfs dfsadmin -refreshUserToGroupsMappings
yarn rmadmin -refreshUserToGroupsMappings
```

3. Проверить сопоставление групп LDAP, выполнив команду *hdfs groups*. Команда отображает группы из LDAP для текущего пользователя. При настроенном сопоставлении групп LDAP разрешения HDFS могут использовать группы, определенные в LDAP для контроля доступа.

- **Ручное создание пользователей и групп в среде Linux**

  Также можно вручную создавать пользователей и группы в среде Linux.

Конфигурация MySQL

При использовании **MySQL** машина для хранения таблиц политики администратора **Ranger** обязательно должна поддерживать транзакции. **InnoDB** – это пример машины, поддерживающей транзакции.

При использовании **Amazon RDS** есть дополнительные требования (*Требования к Amazon RDS*).

Для конфигурации экземпляра для **Ranger** для **MySQL** необходимо выполнить следующие шаги:

1. Для создания баз данных Ranger должен использоваться администратор базы данных MySQL. Для создания пользователя *rangerdba* с паролем *rangerdba* необходимо:

- Войти в систему как пользователь *root* и использовать следующие команды, чтобы создать пользователя *rangerdba* и предоставить ему соответствующие права:

```
CREATE USER 'rangerdba'@'localhost' IDENTIFIED BY 'rangerdba';

GRANT ALL PRIVILEGES ON *.* TO 'rangerdba'@'localhost';

CREATE USER 'rangerdba'@'%' IDENTIFIED BY 'rangerdba';

GRANT ALL PRIVILEGES ON *.* TO 'rangerdba'@'%';

GRANT ALL PRIVILEGES ON *.* TO 'rangerdba'@'localhost' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON *.* TO 'rangerdba'@'%' WITH GRANT OPTION;

FLUSH PRIVILEGES;
```

- Использовать команду *exit* для выхода из MySQL;
- Теперь можно подключиться к базе данных как *rangerdba*, используя следующую команду:

```
mysql -u rangerdba -prangerdba
```

После тестирования входа в систему *rangerdba* использовать команду *exit* для выхода из MySQL.

2. Следующая команда используется для подтверждения, что файл *mysql-connector-java.jar* находится в папке общего доступа Java. Команда должна быть запущена на сервере, на котором установлен сервер Ambari:

```
ls /usr/share/java/mysql-connector-java.jar
```

Если файл находится не в каталоге общего доступа Java, использовать следующую команду для установки соединения:

- RHEL/CentOS/Oracle Linux:

```
yum install mysql-connector-java*
```

- SLES:

```
zypper install mysql-connector-java*
```

3. Использовать следующий формат команды, чтобы установить путь *jdbc/driver/path* на основе местоположения файла *.jar* драйвера MySQL JDBC. Команда должна выполняться на сервере, на котором установлен сервер Ambari:

```
ambari-server setup --jdbc-db={database-type} --jdbc-driver={/jdbc/driver/path}
```

Например:

```
ambari-server setup --jdbc-db=mysql --jdbc-driver=/usr/share/java/mysql-connector-java.jar
```

Конфигурация PostgreSQL

При использовании **Amazon RDS** есть дополнительные требования (*Требования к Amazon RDS*).

Для конфигурации экземпляра для **Ranger** для **PostgreSQL** необходимо выполнить следующие шаги:

1. На хосте PostgreSQL установить соответствующий коннектор PostgreSQL:

- RHEL/CentOS/Oracle Linux:

```
yum install postgresql-jdbc*
```

- SLES:

```
zypper install -y postgresql-jdbc
```

2. Убедиться, что файл *.jar* находится в папке общего доступа Java:

```
ls /usr/share/java/postgresql-jdbc.jar
```

3. Изменить режим доступа файла *.jar* на *644*:

```
chmod 644 /usr/share/java/postgresql-jdbc.jar
```

4. Для создания баз данных Ranger должен использоваться администратор базы данных PostgreSQL. Для создания пользователя *rangerdba* и предоставления ему соответствующих прав следует использовать команду:

```
echo "CREATE DATABASE $dbname;" | sudo -u $postgres psql -U postgres
echo "CREATE USER $rangerdba WITH PASSWORD '$passwd';" | sudo -u $postgres psql -U postgres
echo "GRANT ALL PRIVILEGES ON DATABASE $dbname TO $rangerdba;" | sudo -u $postgres psql -U␣
↪postgres
```

```
```

Где *$postgres* – пользователь Postgres, *$dbname* – имя базы данных PostgreSQL.

5. Использовать следующий формат команды, чтобы установить путь *jdbc/driver/path* на основе местоположения файла *.jar* драйвера PostgreSQL JDBC. Команда должна выполняться на сервере, на котором установлен сервер Ambari:

```
ambari-server setup --jdbc-db={database-type} --jdbc-driver={/jdbc/driver/path}
```

Например:

```
ambari-server setup --jdbc-db=postgres --jdbc-driver=/usr/share/java/postgresql-jdbc.jar
```

6. Выполнить следующую команду:

```
export HADOOP_CLASSPATH=${HADOOP_CLASSPATH}:${JAVA_JDBC_LIBS}:/connector jar path
```

7. Разрешить доступ *Allow Access* для пользователей Ranger:

- изменить *listen_ addresses='localhost'* на *listen_ addresses=''* ('*' = any)*, чтобы прослушивать все IP-адреса в *postgresql.conf*;
- внести следующие изменения пользователям *Ranger db* и *Ranger audit db* в файле *pg_ hba.conf* (Рис.2.1.).



Рис.2.1.: Необходимые изменения пользователям Ranger db и Ranger audit db

8. После редактирования файла *pg_ hba.conf* запустить команду для обновления конфигурации базы данных PostgreSQL:

```
sudo -u postgres /usr/bin/pg_ctl -D $PGDATA reload
```

Например, если файл *pg_ hba.conf* находится в каталоге */var/lib/pgsql/data*, значением *$PGDATA* является */var/lib/pgsql/data*.

Конфигурация Oracle

При использовании **Amazon RDS** есть дополнительные требования (*Требования к Amazon RDS*).

Для конфигурации экземпляра для **Ranger** для **Oracle** необходимо выполнить следующие шаги:

1. На узле Oracle установить соответствующий JDBC-файл *.jar*:

- Загрузить драйвер Oracle JDBC (OJDBC)
- Для Oracle Database 11g: выбрать Oracle Database 11g Release 2 drivers > ojdbc6.jar
- Для Oracle Database 12c: выбрать Oracle Database 12c Release 1 driver > ojdbc7.jar
- Скопировать файл *.jar* в папку общего доступа Java. Например, *cp ojdbc7.jar /usr/share/java/*

- Убедиться, что .jar-файл имеет соответствующие разрешения:

```
chmod 644 /usr/share/java/ojdbc7.jar
```

2. Для создания баз данных Ranger должен использоваться администратор базы данных Oracle.

Для создания пользователя *RANGERDBA* и предоставления ему прав с помощью SQL*Plus – утилиты администрирования базы данных Oracle, следует использовать команду:

```
# sqlplus sys/root as sysdba
CREATE USER $RANGERDBA IDENTIFIED BY $RANGERDBAPASSWORD;
GRANT SELECT_CATALOG_ROLE TO $RANGERDBA;
GRANT CONNECT, RESOURCE TO $RANGERDBA;
QUIT;
```

3. Использовать следующий формат команды, чтобы установить путь *jdbc/driver/path* на основе местоположения файла *.jar* драйвера Oracle JDBC. Команда должна выполняться на сервере, на котором установлен сервер Ambari:

```
ambari-server setup --jdbc-db={database-type} --jdbc-driver={/jdbc/driver/path}
```

Например:

```
ambari-server setup --jdbc-db=oracle --jdbc-driver=/usr/share/java/ojdbc6.jar
```

Требования к Amazon RDS

Ranger требует наличия реляционной базы данных в качестве хранилища политик. Существуют дополнительные требования для баз данных на основе **Amazon RDS** из-за специфичности настроек и управления.

- **MySQL/MariaDB**

Во время установки **Ranger** необходимо изменить переменную *log_bin_trust_function_creators* на значение *1*. Через панель управления RDS Dashboard > Parameter group (в левой части страницы):

- Установить переменную MySQL Server *log_bin_trust_function_creators* в значение *1*.
- (Опционально) после завершения установки Ranger сбросить значение параметра *log_bin_trust_function_creators* в исходное значение (требование к значению переменной относится только на время установки Ranger).

Дополнительная информация:

- Stratalux: Why You Should Always Use a Custom DB Parameter Group When Creating an RDS Instance
- AWS Documentation>Amazon RDS DB Instance Lifecycle » Working with DB Parameter Groups
- MySQL 5.7 Reference Manual >Binary Logging of Stored Programs

- **PostgreSQL**

Пользователь базы данных **Ranger** на сервере **Amazon RDS PostgreSQL Server** должен быть создан до установки **Ranger** и ему должна быть предоставлена роль *CREATEDB*.

1. Используя основную учетную запись пользователя (заведенную при создании экземпляра RDS PostgreSQL), войти в Amazon RDS PostgreSQL Server и выполнить команды:

```
CREATE USER $rangerdbuser WITH LOGIN PASSWORD 'password'

GRANT $rangerdbuser to $postgresroot
```

Где *$postgresroot* – это основная учетная запись пользователя RDS PostgreSQL (например, *postgresroot*), а *$rangerdbuser* – имя пользователя базы данных Ranger (например: *rangeradmin*).

2. Если используется Ranger KMS, выполнить следующие команды:

```
CREATE USER $rangerkmsuser WITH LOGIN PASSWORD 'password'

GRANT $rangerkmsuser to $postgresroot
```

Где *$postgresroot* – это основная учетная запись пользователя RDS PostgreSQL (например, *postgresroot*), а *$rangerkmsuser* – имя пользователя Ranger KMS (например, *rangerkms*).

- **Oracle**

Из-за ограничений в Amazon RDS создание пользователя базы данных **Ranger** и табличного пространства, а так же предоставление пользователю **Ranger** необходимых привилегий выполняется вручную.

1. Используя основную учетную запись пользователя (заведенную при создании экземпляра RDS Oracle), войти в RDS Oracle Server и выполнить команды:

```
create user $rangerdbuser identified by “password”;
GRANT CREATE SESSION,CREATE PROCEDURE,CREATE TABLE,CREATE VIEW,CREATE SEQUENCE,CREATE PUBLIC␣
↪SYNONYM,CREATE ANY SYNONYM,CREATE TRIGGER,UNLIMITED Tablespace TO $rangerdbuser;
create tablespace $rangerdb datafile size 10M autoextend on;
alter user $rangerdbuser DEFAULT Tablespace $rangerdb;
```

Где *$rangerdb* – это фактическое имя базы данных Ranger (например, *ranger*), а *$rangerdbuser* – имя пользователя Ranger (например: *rangeradmin*).

2. Если используется Ranger KMS, выполнить следующие команды:

```
create user $rangerdbuser identified by “password”;
GRANT CREATE SESSION,CREATE PROCEDURE,CREATE TABLE,CREATE VIEW,CREATE SEQUENCE,CREATE PUBLIC␣
↪SYNONYM,CREATE ANY SYNONYM,CREATE TRIGGER,UNLIMITED Tablespace TO $rangerkmsuser;
create tablespace $rangerkmsdb datafile size 10M autoextend on;
alter user $rangerkmsuser DEFAULT Tablespace $rangerkmsdb;
```

Где *$rangerkmsdb* – это фактическое имя базы данных Ranger (например: *rangerkms*), а *$rangerkmsuser* – имя пользователя Ranger (например: *rangerkms*).

2.2 Установка Ranger

Установка **Ranger** с помощью **Ambari** заключается в три этапа:

- *Запуск инсталляции*
- *Настройка сервисов*
- *Завершение установки*

Смежные темы:

- *Расширенные настройки пользователей*
- *Настройка Ranger для LDAP SSL*
- *Настройка пользователей без использования учетных данных DBA*
- *Обновление паролей администратора Ranger*
- *Включение плагинов Ranger*

2.2.1 Запуск инсталляции

Запуск инсталляции осуществляется по следующему сценарию:

1. Войти в кластер Ambari с помощью назначенных учетных данных пользователя. При этом отображается главная страница панели инструментов Ambari (Рис.2.2.).

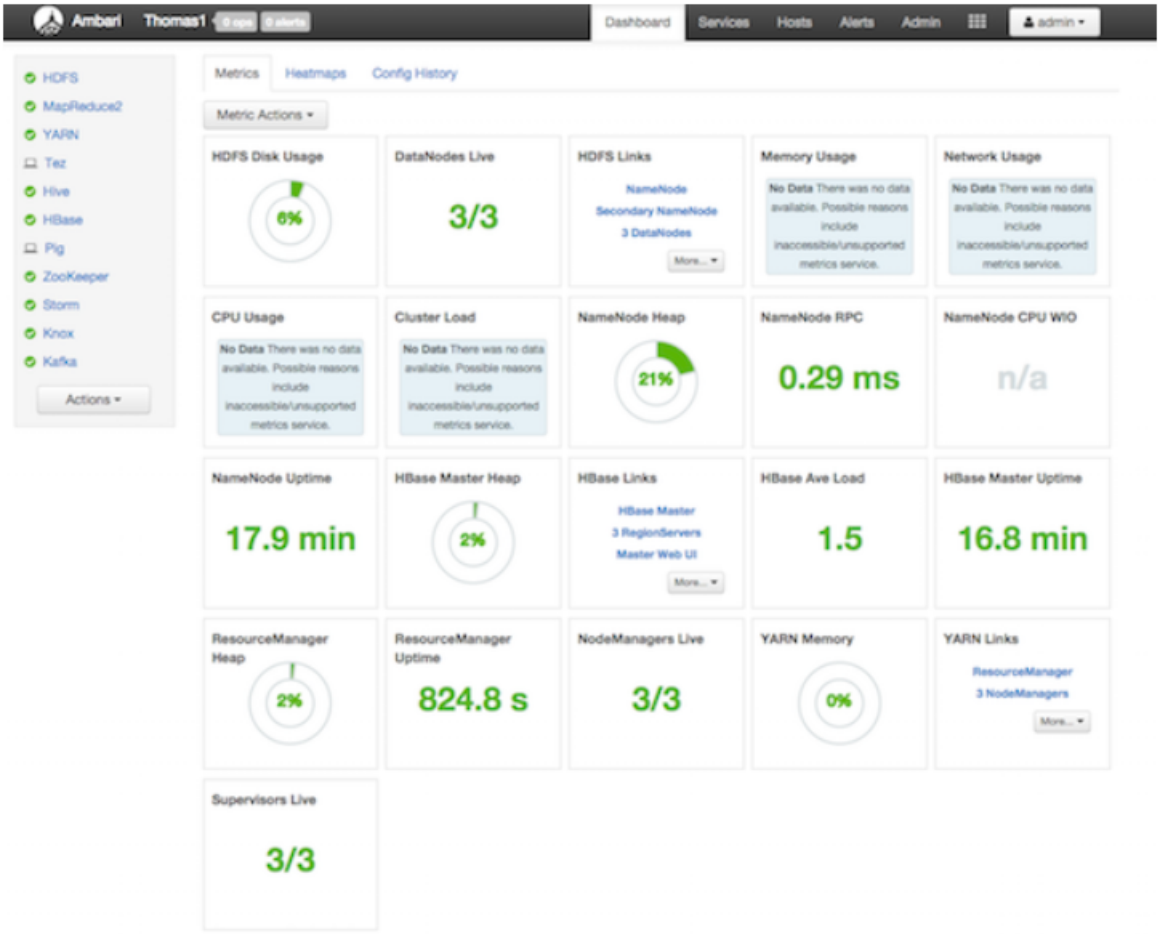


Рис.2.2.: Главная страница Ambari

2. В левом меню навигации выбрать пункты меню “Actions > Add Service” (Рис.2.3.).
3. На открывшейся странице “Choose Services” выбрать *Ranger* и нажать кнопку *Next* (Рис.2.4.).
4. Открывается страница “Ranger Requirements”. Необходимо убедиться, что выполнены все требования к установке, установить флажок *I have met all the requirements above* и нажать кнопку *Proceed* (Рис.2.5.).
5. Далее на открывшейся странице “Assign Masters” необходимо выбрать хост, на котором будет установлен Ranger Admin (Рис.2.6.). Этот хост должен иметь доступ администратора базы данных к хосту Ranger DB и User Sync. На приведенном рисунке показано, что службы Ranger Admin и Ranger User Sync будут установлены на основном узле кластера (*c6401.ambari.apache.org*). Следует запомнить хост администратора Ranger для использования на последующих этапах установки. Нажать кнопку *Next* для продолжения.
6. Открывается страница “Customize Services” (Рис.2.7.). Настройки сервисов описаны в следующем разделе (*Настройка сервисов*).

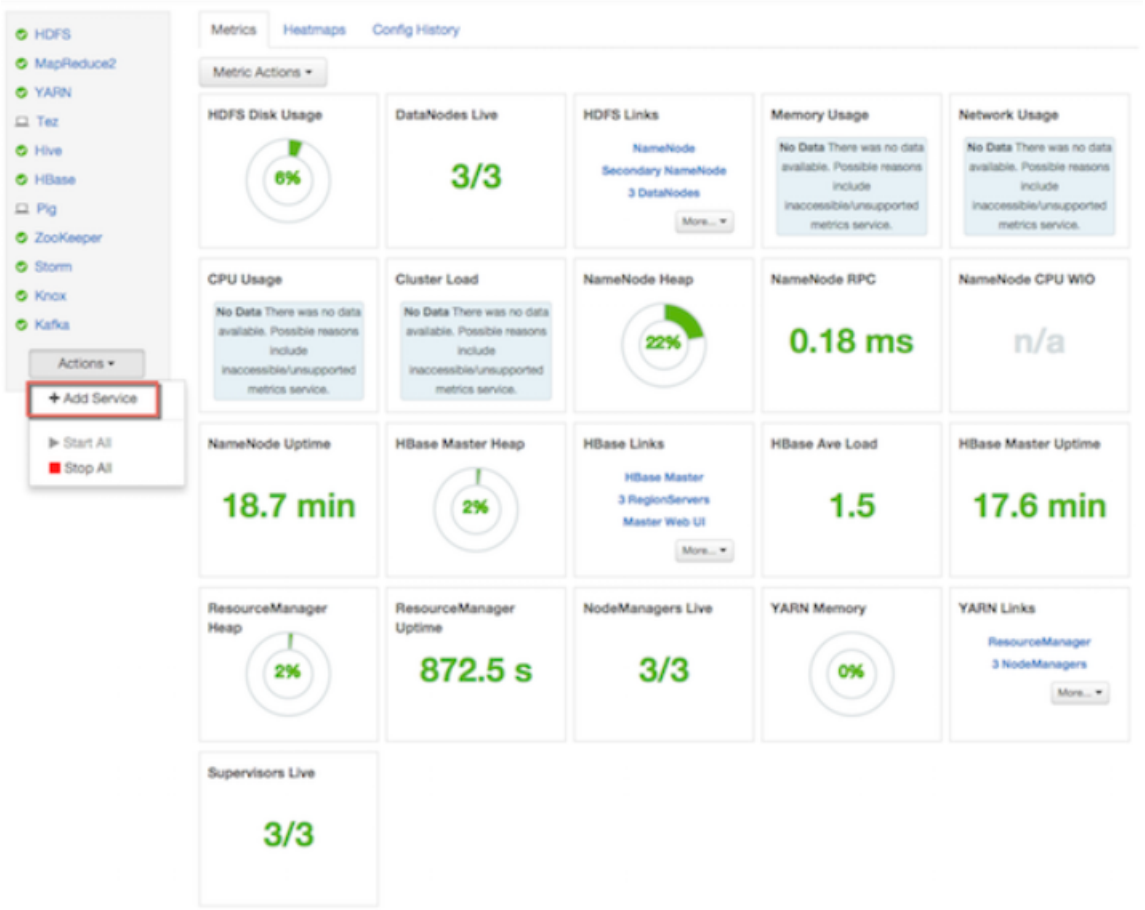


Рис.2.3.: Действие – Добавить сервис

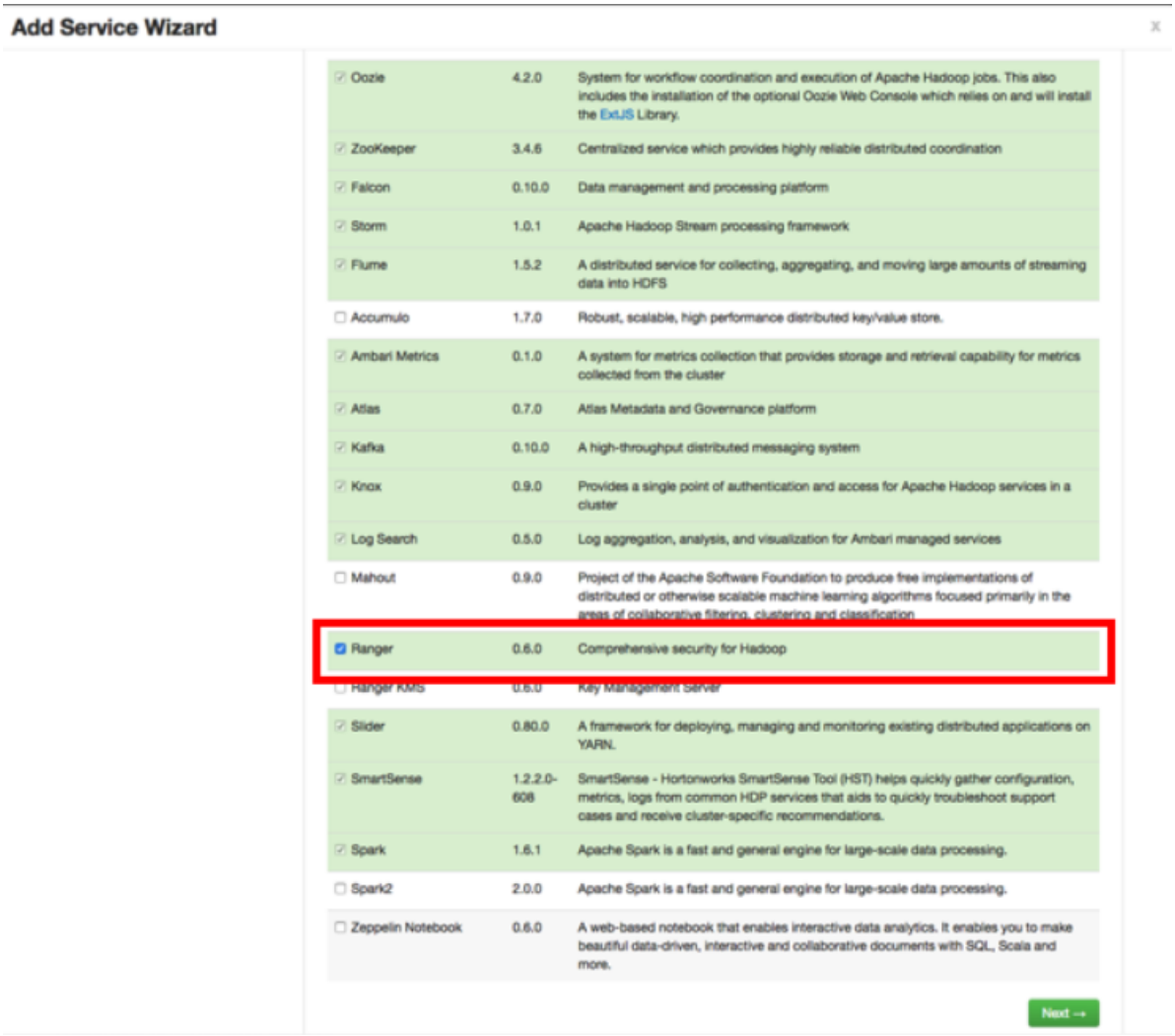


Рис.2.4.: Добавление сервиса

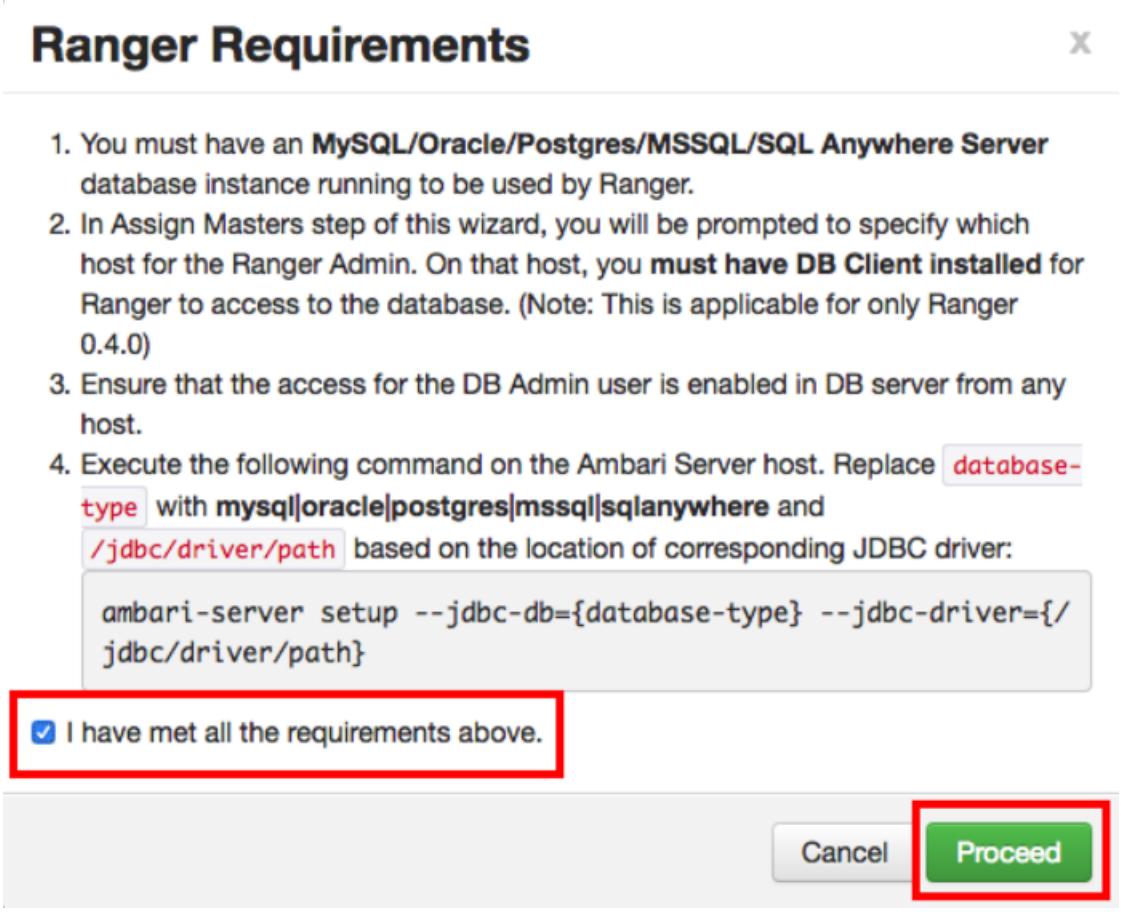


Рис.2.5.: Требования Ranger

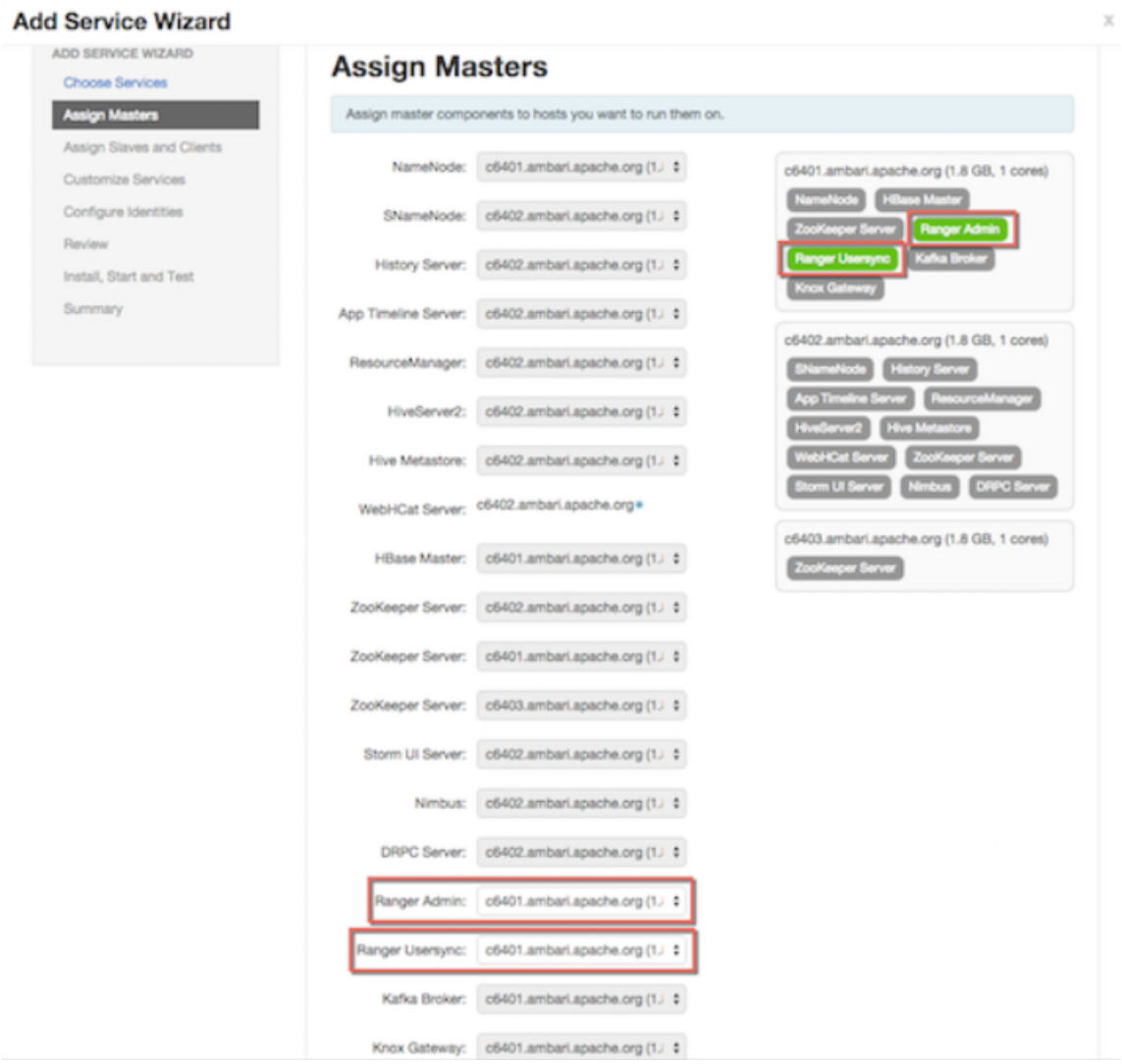


Рис.2.6.: Выбор хоста для установки Ranger Admin

2.2.2 Настройка сервисов

Следующим шагом в процессе установки **Ranger** является задание настроек на странице “Customize Services” (Рис.2.7.):

- *Ranger Admin*
- *Ranger Audit*
- *Ranger User Sync*
- *Ranger Tagsync*
- *Ranger Authentication*

Ranger Admin

Настройка администратора **Ranger** выполняется в следующем порядке:

1. На странице “Customize Services” выбрать вкладку “Ranger Admin” и в раскрывающемся списке “DB Flavor” выбрать тип базы данных, используемый с Ranger (Рис.2.7.).

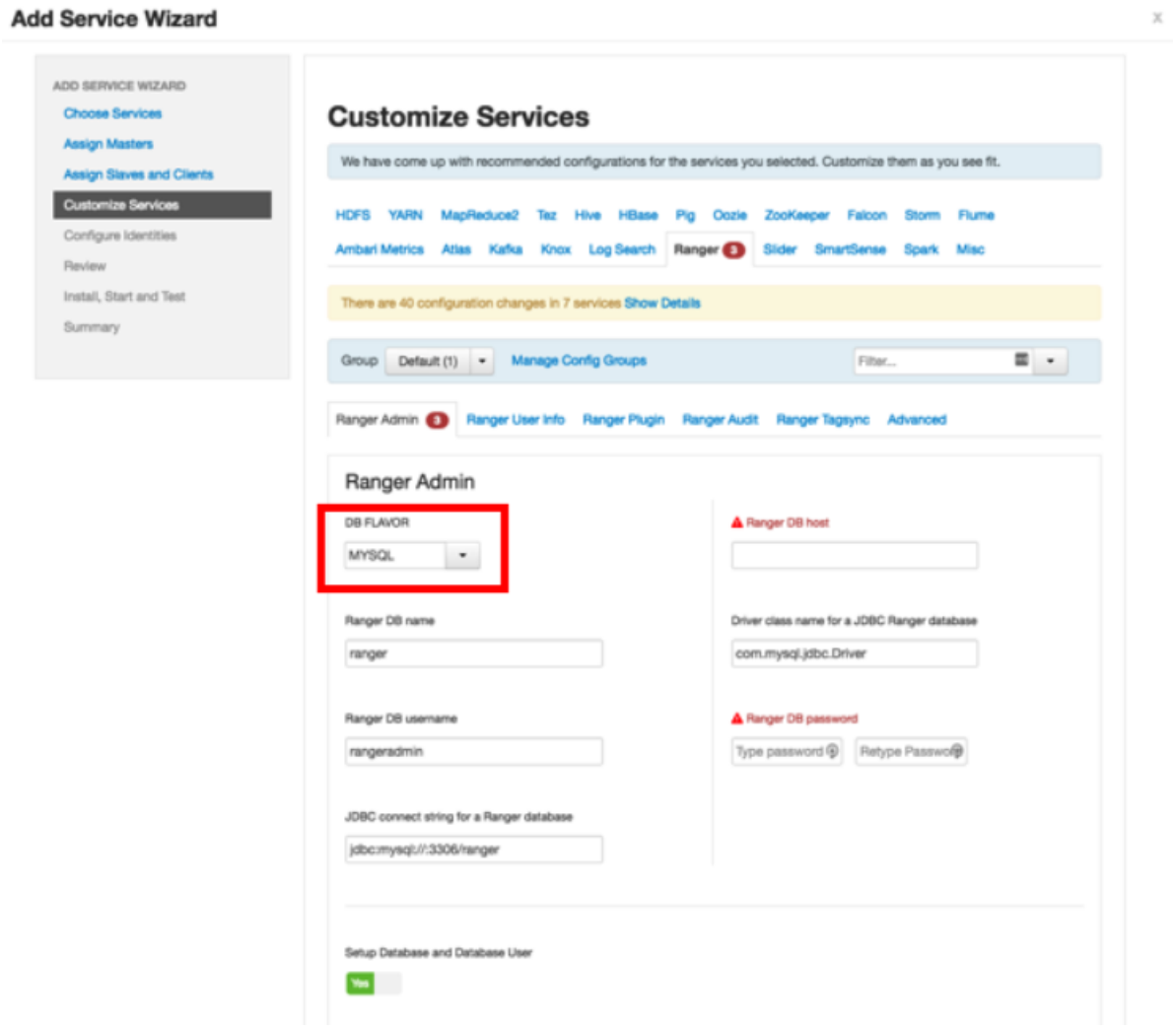


Рис.2.7.: Выбор типа базы данных

2. Ввести адрес сервера базы данных в поле “Ranger DB Host” в соответствии с таблицей.

Таблица2.1.: Ranger DB Host

DB Flavor	Host	Пример
MySQL	<HOST[:PORT]>	c6401.ambari.apache.org или c6401.ambari.apache.org:3306
Oracle	<HOST:PORT:SID>	c6401.ambari.apache.org:1521:ORCL
Oracle	<HOST:PORT/Service>	c6401.ambari.apache.org:1521/XE
PostgreSQL	<HOST[:PORT]>	c6401.ambari.apache.org или c6401.ambari.apache.org:5432
MS SQL	<HOST[:PORT]>	c6401.ambari.apache.org или c6401.ambari.apache.org:1433
SQLA	<HOST[:PORT]>	c6401.ambari.apache.org или c6401.ambari.apache.org:2638

3. Поле “Ranger DB name” – имя базы данных Ranger Policy, то есть *Ranger_db*.

Important: При использовании Oracle указать имя табличного пространства Oracle

4. Поле “Driver class name for a JDBC Ranger database” – имя класса драйвера для базы данных JDBC Ranger – создается автоматически на основе выбранного типа в поле “DB Flavor”. В приведенной таблице перечислены настройки класса драйвера по умолчанию (в настоящее время Ranger не поддерживает сторонний драйвер JDBC).

Таблица2.2.: Driver Class Name

DB Flavor	Driver class name для JDBC Ranger
MySQL	com.mysql.jdbc.Driver
Oracle	oracle.jdbc.driver.OracleDriver
PostgreSQL	org.postgresql.Driver
MS SQL	com.microsoft.sqlserver.jdbc.SQLServerDriver
SQLA	sap.jdbc4.sqlanywhere.IDriver

5. В поля “Ranger DB username” и “Ranger DB Password” необходимо ввести имя пользователя и пароль для сервера базы данных Ranger. В таблице описаны более детальные настройки. Можно использовать базу данных MySQL, установленную с Ambari, или внешнюю БД: MySQL, Oracle, PostgreSQL, MS SQL или SQL Anywhere.

Таблица2.3.: Пользователь и пароль Ranger DB

	Ranger DB username	Ranger DB password
Описание	Имя пользователя для базы данных Policy	Пароль для пользователя базы данных Ranger Policy
Значение по умолчанию	rangeradmin	
Пример значения	rangeradmin	PassWORd
Обязательность заполнения	Да	Да

6. Строка подключения JDBC – в настоящее время установщик Ambari создает строку соединения JDBC, используя формат *jdbc:oracle:thin:@//host:port/db_name*. Необходимо заменить строку подключения:

- **MySQL** – синтаксис: *jdbc:mysql://DB_HOST:PORT/db_name*, пример значения:

```
jdbc:mysql://c6401.ambari.apache.org:3306/ranger_db
```

- **Oracle SID** – синтаксис: *jdbc:oracle:thin:@DB_HOST:PORT:SID*, пример значения:

```
jdbc:oracle:thin:@c6401.ambari.apache.org:1521:ORCL
```

- **Oracle Service Name** – синтаксис: *jdbc:oracle:thin:@//DB_HOST[:PORT][/ServiceName]*, пример значения:

```
jdbc:oracle:thin:@//c6401.ambari.apache.org:1521/XE
```

- **PostgreSQL** – синтаксис: *jdbc:postgresql://DB_HOST/db_name*, пример значения:

```
jdbc:postgresql://c6401.ambari.apache.org:5432/ranger_db
```

- **MS SQL** – синтаксис: *jdbc:sqlserver://DB_HOST;databaseName=db_name*, пример значения:

```
jdbc:sqlserver://c6401.ambari.apache.org:1433;databaseName=ranger_db
```

- **SQLA** – синтаксис: *jdbc:sqlanywhere:host=DB_HOST;database=db_name*, пример значения:

```
jdbc:sqlanywhere:host=c6401.ambari.apache.org:2638;database=ranger_db
```

7. Поле “Setup Database and Database User”:

- при установке значения *Yes* имя и пароль администратора базы данных необходимо будет предоставить, как описано на шаге 8. Ranger не сохраняет имя и пароль DBA после установки. Таким образом можно очистить эти значения в пользовательском интерфейсе Ambari после завершения настройки Ranger;
- установка значения *No* означает отказ от предоставления данных учетной записи DBA установщику Ambari Ranger. Процесс установки Ranger продолжится без предоставления этих данных. В таком случае необходимо выполнить настройку пользователя базы данных системы, как описано в разделе *Настройка пользователей без использования учетных данных DBA*, а затем приступить к установке. При этом пользовательский интерфейс по-прежнему требует ввода имени и пароля для продолжения, тогда можно ввести любые значения (значения не обязательно должны быть фактическим именем и паролем администратора).

8. “Database Administrator (DBA) username” и “Database Administrator (DBA) password” задаются при установке сервера баз данных. Если эти сведения отсутствуют, необходимо обратиться к администратору базы данных, установившему сервер.

Таблица2.4.: Настройки учетных данных DBA

	DBA username	DBA password
Описание	Пользователь базы данных Ranger, обладающий правами администратора для создания схем баз данных и пользователей	Пароль пользователя базы данных Ranger
Значение по умолчанию	root	
Пример значения	root	root
Обязательность заполнения	Да	Да

Если роль пользователя root Oracle DB – *SYSDBA*, необходимо указать это в параметре имени администратора базы данных. Например, если имя пользователя DBA – *orcl_root*, следует указать *orcl_root AS SYSDBA*.

Как упомянуто на предыдущем шаге, если “Setup Database and Database User” установлено в положение *No*, имя и пароль DBA могут все еще требоваться для продолжения установки Ranger.

На следующих рисунках показаны примеры настроек БД для каждого типа базы данных Ranger (Рис.2.8., Рис.2.9., Рис.2.10., Рис.2.11., Рис.2.12., Рис.2.13.).

Important: Чтобы проверить настройки БД, следует нажать “Test Connection”. Если база данных Ranger не была предварительно установлена, тестовое соединение завершится неудачно даже при правильной конфигурации

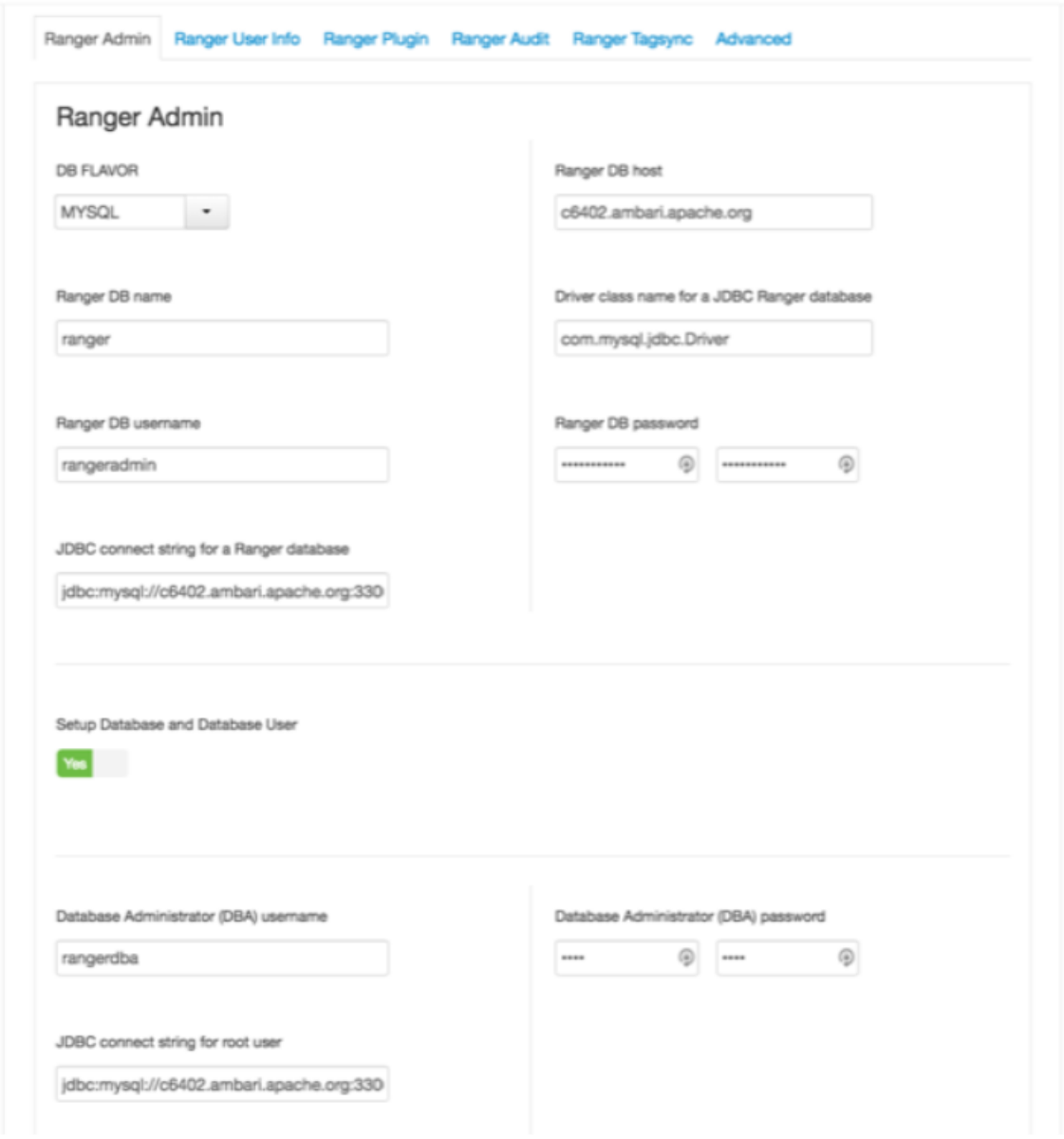


Рис.2.8.: MySQL

Ranger Audit

Apache Ranger использует **Apache Solr** для хранения журналов аудита и обеспечивает поиск пользовательского интерфейса через них. **Solr** необходимо установить и настроить перед инсталляцией **Ranger**

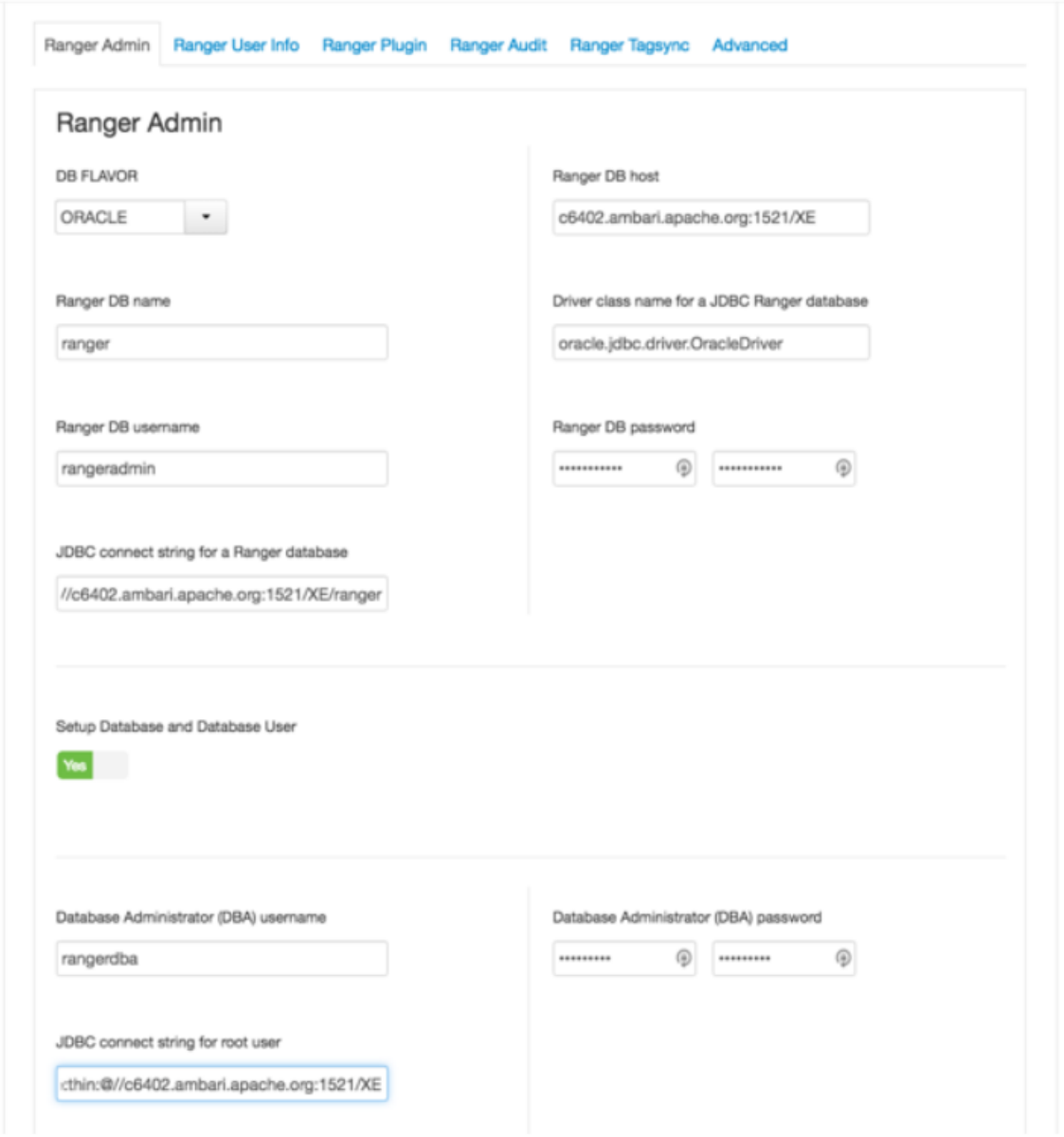


Рис.2.9.: Oracle Service Name

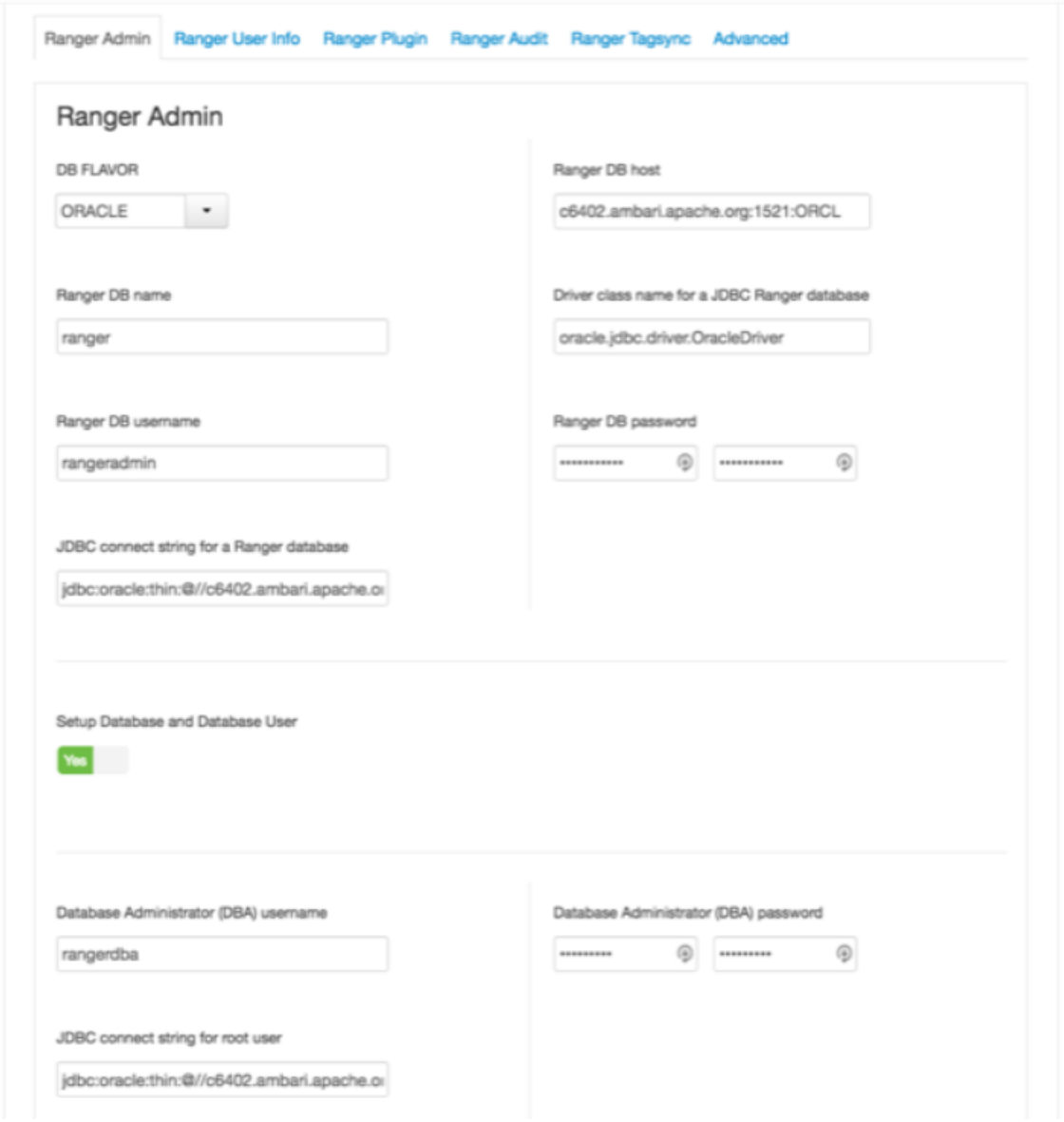


Рис.2.10.: Oracle SID

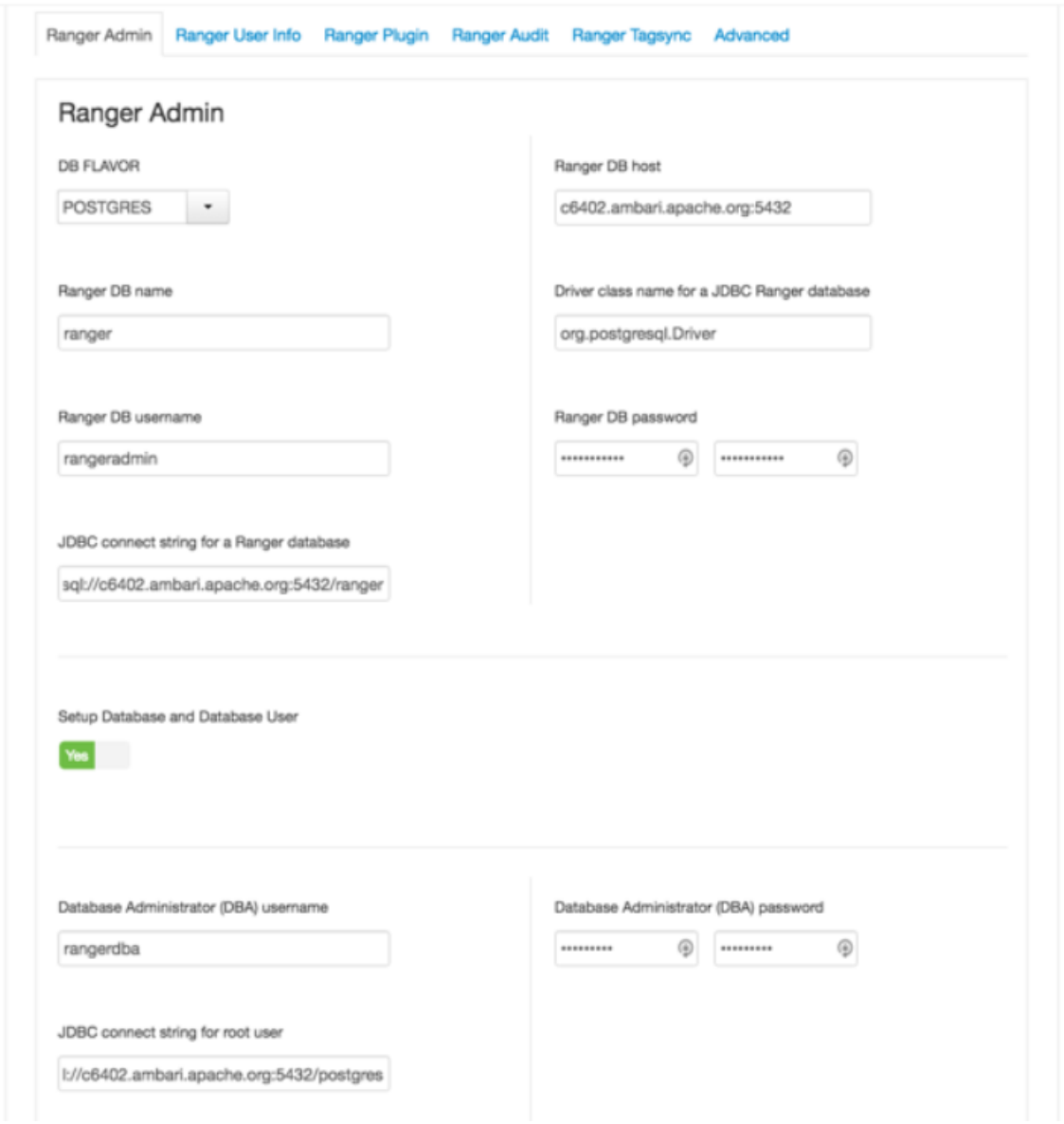


Рис.2.11.: PostgreSQL

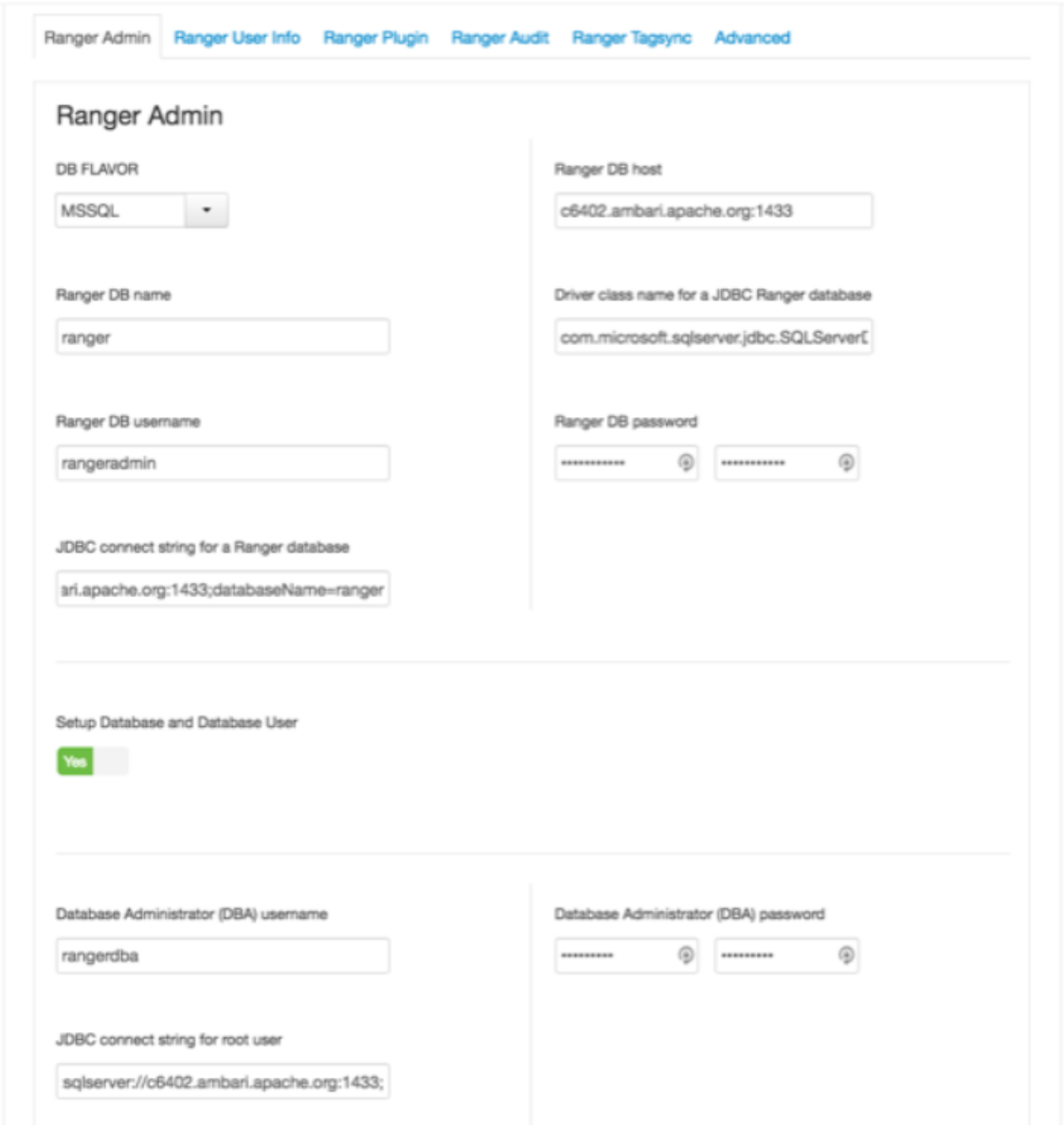


Рис.2.12.: MS SQL

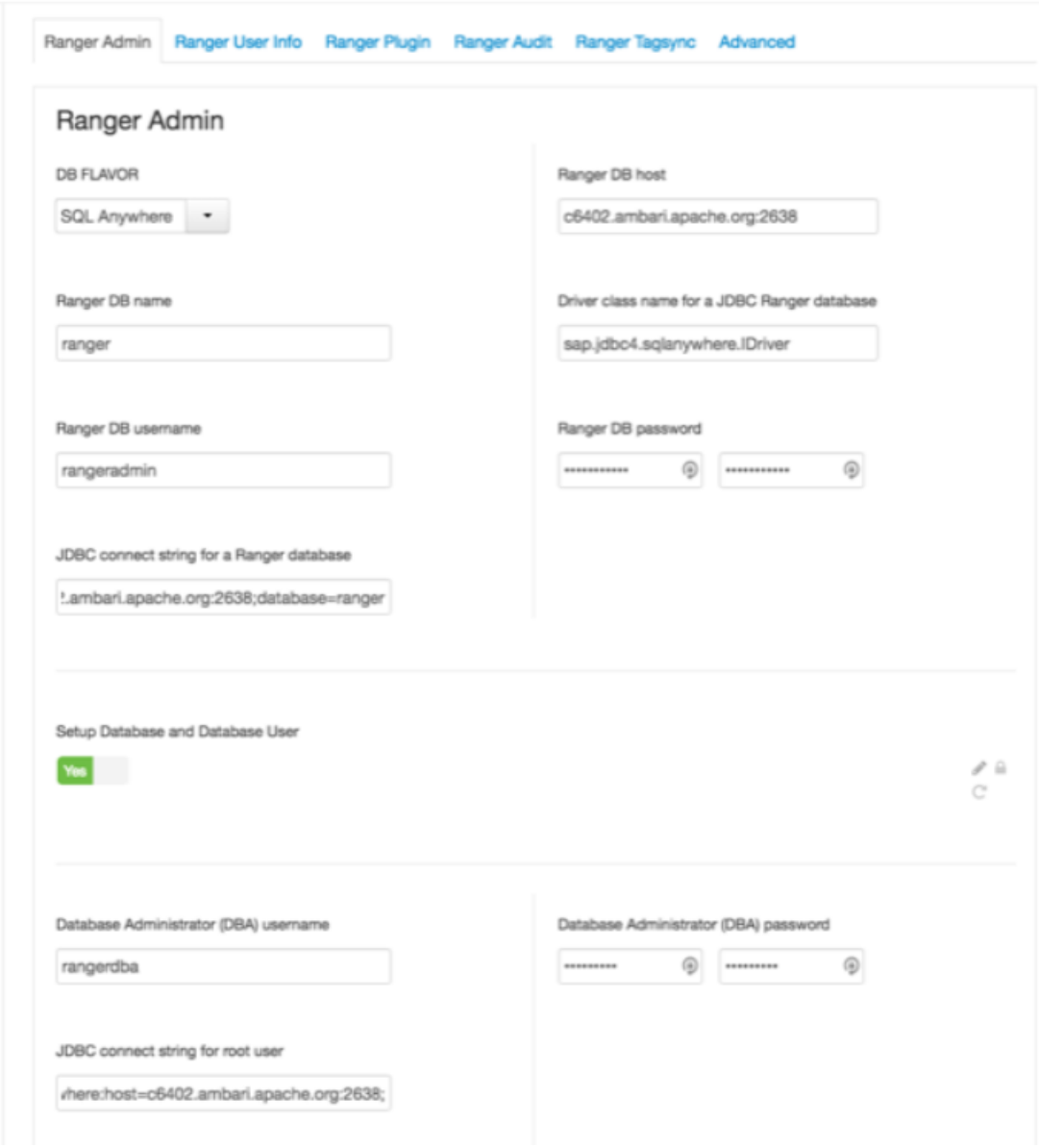


Рис.2.13.: SQL Anywhere

Admin или любого из плагинов компонента **Ranger**. Конфигурация по умолчанию для **Ranger Audits** в **Solr** использует общий экземпляр **Solr**, предоставляемый сервисом **Ambari Infra**. **Solr** – это и память, и процессор. Если продуктивная система имеет большой объем запросов доступа, необходимо убедиться, что хост **Solr** имеет достаточную память, процессор и дисковое пространство.

SolrCloud является предпочтительной установкой для использования **Ranger**. **SolrCloud**, разворачиваемый с сервисом **Ambari Infra**, представляет собой масштабируемую архитектуру, которая может работать как единый узел или кластер с несколькими узлами. Он имеет дополнительные функции, такие как репликация и сегментирование, что полезно для высокой доступности (HA) и масштабируемости.

Следует планировать развертывание на основе размера кластера. Поскольку записи аудита могут значительно увеличиваться, важно иметь не менее *1 ТБ* свободного места, где **Solr** будет хранить данные индекса. Необходимо предоставить процессу **Solr** как можно больше памяти (хорошо работает с *32 ГБ* оперативной памяти). Настоятельно рекомендуется использовать **SolrCloud** по меньшей мере с двумя узлами **Solr**, работающими на разных серверах с включенной репликацией. **SolrCloud** также требует **Apache ZooKeeper**.

1. На странице “Customize Services” выбрать вкладку “Ranger Audit” (см. Рис.2.7.).

Рекомендуется хранить аудиты в Solr и HDFS. Обе эти опции заданы по умолчанию (установлены в положение *ON*). Solr предоставляет возможность индексирования и поиска по самым последним журналам, в то время как HDFS используется как более постоянное и долгосрочное хранилище. По умолчанию Solr используется для индексации журналов аудита за предшествующие 30 дней.

2. В блоке “Audit to Solr” в поле “SolrCloud” установить значение *ON* для активирования SolrCloud (Рис.2.14.). При этом настройки конфигурации SolrCloud будут загружены автоматически.

Ranger User Sync

В разделе описывается настройка **Ranger User Sync** для **UNIX** и **LDAP/AD**.

- *Тест-драйв Ranger Usersync*
- *Настройка синхронизации пользователей Ranger для UNIX*
- *Настройка синхронизации пользователя Ranger для LDAP/AD*
- *Автоматическое назначение роли ADMIN/KEYADMIN для внешних пользователей*

Тест-драйв Ranger Usersync

Перед применением изменений в **usersync** рекомендуется выполнить тестовый запуск, чтобы пользователи и группы извлекались должным образом. Для тестового запуска загрузки данных User и Group в **Ranger** перед фиксацией изменений необходимо:

1. Установить параметр в значение *ranger.usersync.policymanager.mockrun=true*. Он находится в *Ambari> Ranger> Configs> Advanced> Advanced ranger-ugsync-site*
2. Проверить пользователей и группы для загрузки в Ranger: *tail -f /var/log/ranger/usersync/usersync.log*
3. После подтверждения того, что пользователи и группы будут извлечены по назначению, установить *ranger.usersync.policymanager.mockrun=false* и перезапустить Ranger Usersync.

Эти действия приводят к синхронизации пользователей, отображаемых в журнале **usersync**, с базой данных **Ranger**.

Настройка синхронизации пользователей Ranger для UNIX

Для настройки **Ranger User Sync** для **UNIX** необходимо выполнить следующий порядок действий:

1. На странице “Customize Services” выбрать вкладку “Ranger User Info” (Рис.2.15.);

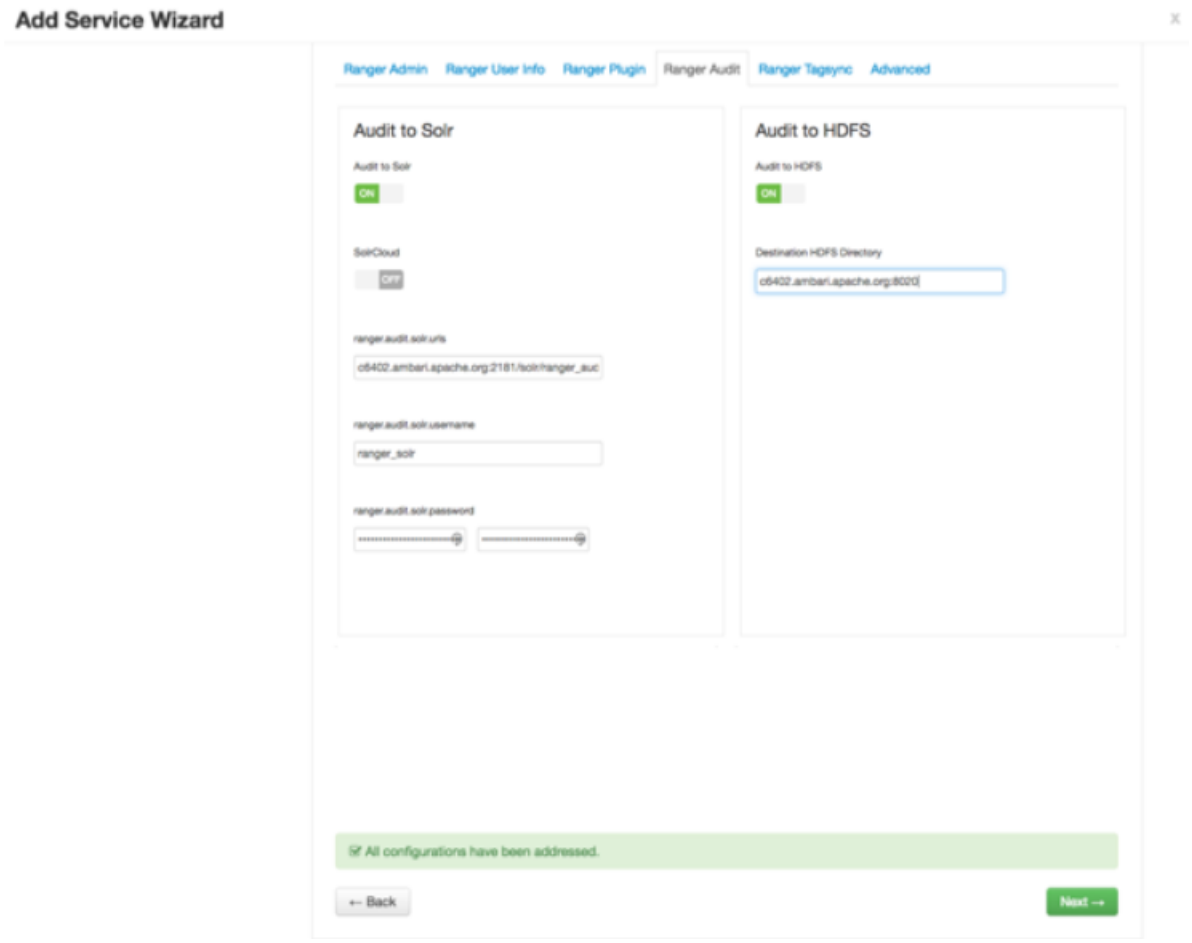


Рис.2.14.: Audit to Solr

2. В разделе “Enable User Sync” установить значение *Yes*;
3. В раскрывающемся списке “Sync Source” выбрать *UNIX*, а затем установить свойства, описание которых приведено в таблице.

Таблица2.5.: Свойства UNIX User Sync

Свойство	Описание	Значение по умолчанию
Sync Source	Синхронизировать пользователей только выше указанно ID	500
Password File	Расположение файла паролей на сервере Linux	/etc/passwd
Group File	Расположение файла групп на сервере Linux	/etc/group

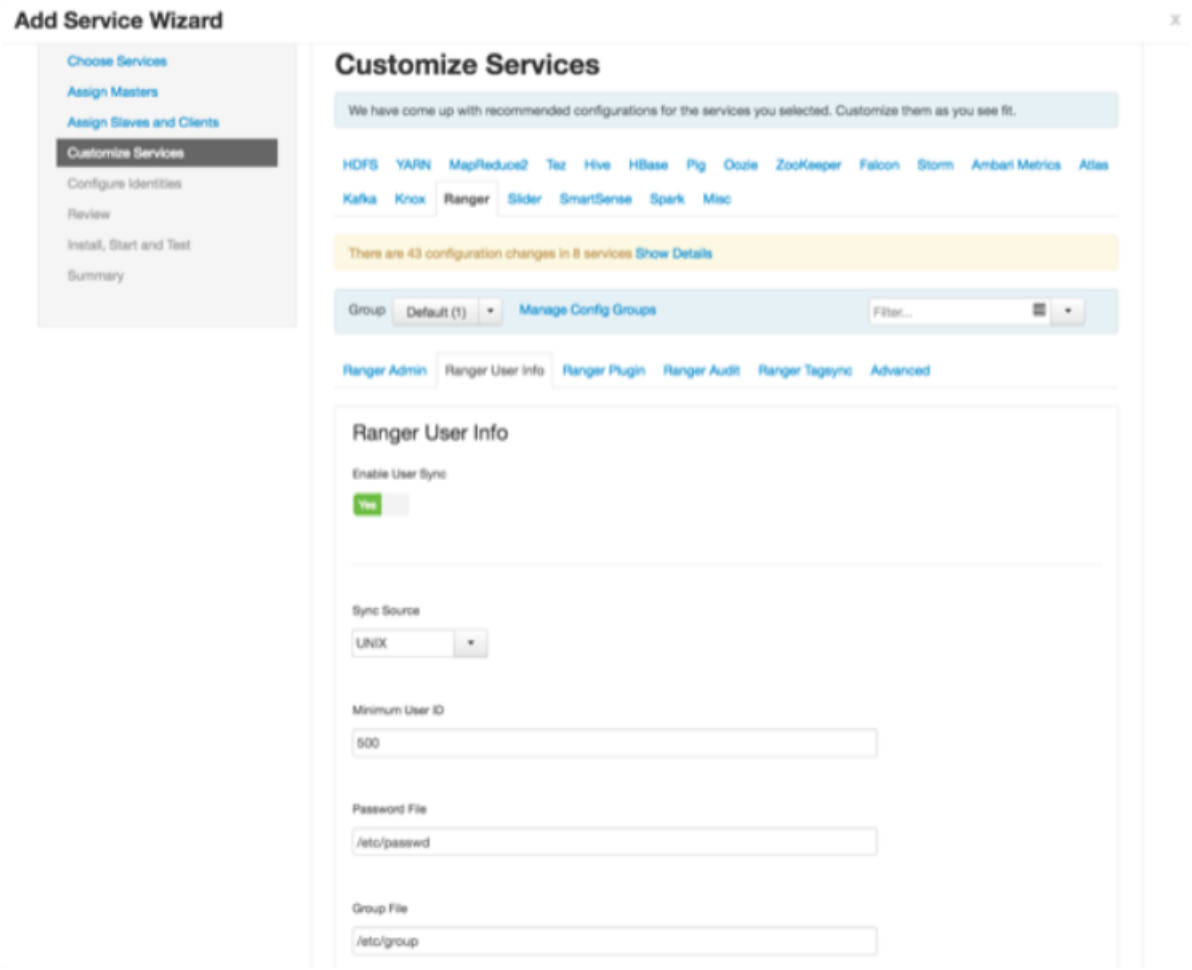


Рис.2.15.: Настройка Ranger User Info для UNIX

Настройка синхронизации пользователя Ranger для LDAP/AD

Для обеспечения принудительной авторизации на уровне групп **LDAP/AD** в **Hadoop** необходимо настроить сопоставление групп Hadoop для LDAP/AD.

Для настройки **Ranger User Sync** для **LDAP/AD** необходимо выполнить следующий порядок действий:

1. На странице “Customize Services” выбрать вкладку “Ranger User Info” (Рис.2.16.);
2. В разделе “Enable User Sync” установить значение *Yes*;
3. В раскрывающемся списке “Sync Source” выбрать *LDAP/AD*, а затем установить свойства:

- **LDAP/AD URL** – Добавление URL в зависимости от источника синхронизации LDAP/AD.
  - Значение по умолчанию – *ldap://{host}:{port}*
  - Пример значения – *ldap://ldap.example.com:389* или *ldaps://ldap.example.com:636*
- **Bind Anonymous** – Если выбрано значение *Yes*, Bind User и Bind User Password не требуются.
  - Значение по умолчанию – *NO*
- **Bind User** – Расположение файла групп на сервере Linux.
  - Значение по умолчанию – Полное distinguished name (DN), включая common name (CN), учетной записи пользователя LDAP/AD с правами поиска пользователей. Используется для запроса пользователей и групп.
  - Пример значения – *cn=admin,dc=example,dc=com* или *admin@example.com*
- **Bind User Password** – Пароль Bind User.
- **Incremental Sync** – Если выбрано *Yes*, Ranger Usersync сохраняет последнюю временную метку всех объектов, которые были синхронизированы ранее, и использует эту метку времени для выполнения следующей синхронизации. Затем Usersync использует механизм опроса для выполнения инкрементной синхронизации с помощью атрибутов LDAP *uSNChanged* (для AD) или *modifytimestamp* (для LDAP). Включение инкрементной синхронизации в первый раз приводит к полной синхронизации; последующие операции синхронизации будут инкрементальными. Когда включена инкрементная синхронизация, групповая синхронизация (на вкладке “Group Configs”) является обязательной. Рекомендуется для крупных развертываний.
  - Значение по умолчанию – Для обновления: *No*; для инсталляции: *Yes*.
  - Пример значения – *Yes*

4. На вкладке “User Configs” установить свойства (Рис.2.17.):

- **Group User Map Sync** – Синхронизация определенных групп для пользователей.
  - Значение по умолчанию – *Yes*
  - Пример значения – *Yes*
- **Username Attribute** – Атрибут имени пользователя LDAP.
  - Пример значения – *sAMAccountName* для AD, *uid* или *cn* для OpenLDAP
- **User Object Class** – Класс объекта для идентификации записей пользователя.
  - Значение по умолчанию – *person*
  - Пример значения – *top*, *person*, *organizationalPerson*, *user* или *posixAccount*
- **User Search Base** – Поиск базы для пользователей. Ranger может искать несколько подразделений в AD. Модуль Ranger UserSync выполняет поиск пользователей по каждому настроенному подразделению и добавляет всех пользователей в один список. После того как все подразделения будут обработаны, членство в группе пользователя вычисляется на основе поиска группы.
  - Пример значения – *cn=users,dc=example,dc=com;ou=example1,ou=example2*
- **User Search Filter** – Дополнительный фильтр, ограничивающий пользователей, выбранных для синхронизации.

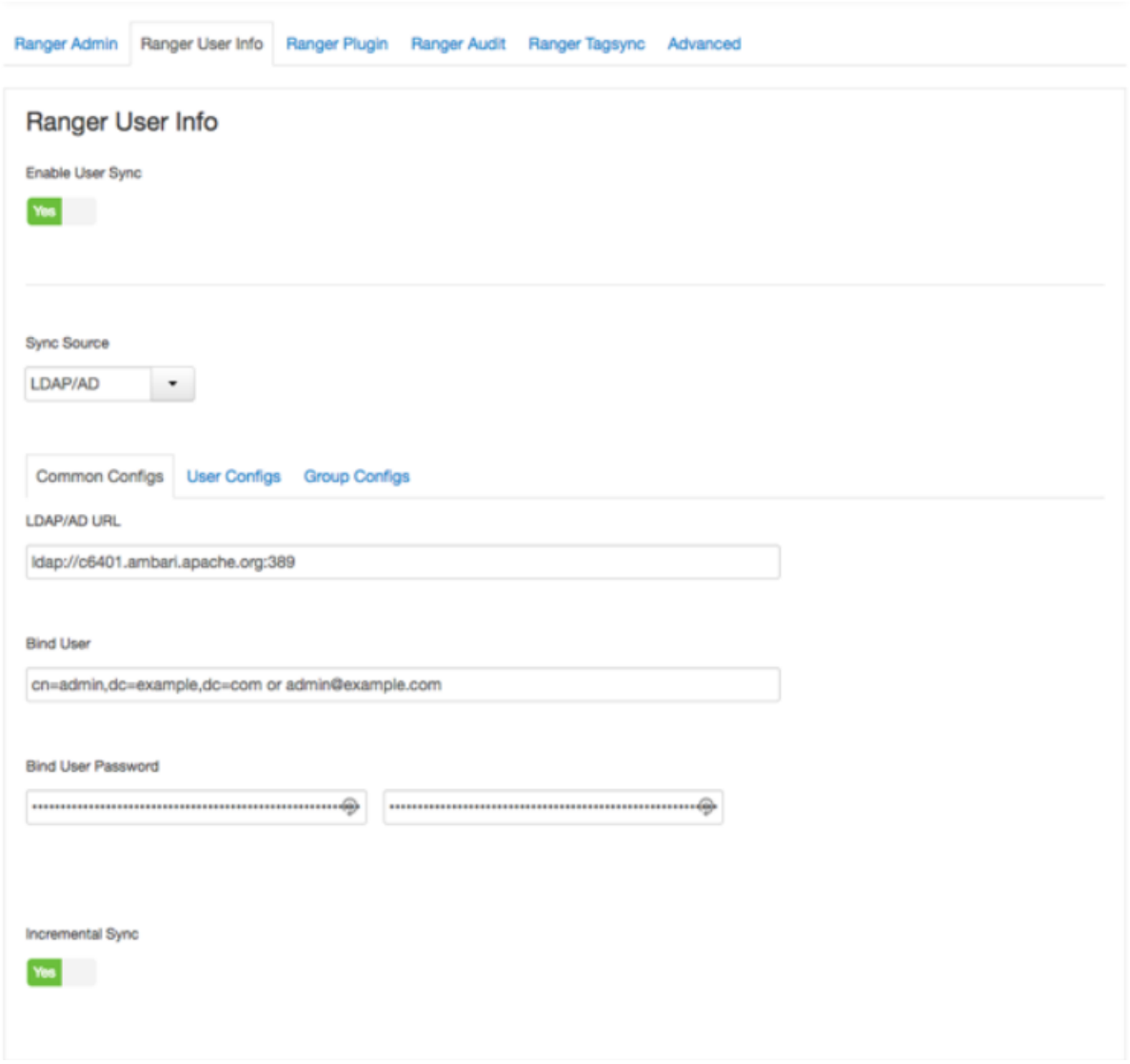


Рис.2.16.: Настройка Ranger User Info для LDAP/AD

  - Пример значения – Для извлечения всех пользователей: cn=*. Для извлечения всех пользователей, которые являются членами groupA или groupB: *(|(memberof=CN=GroupA,OU=groups,DC=example,DC=com)(memberof=CN=GroupB,OU=groups,DC=example,DC=com))*

- **User Search Scope** – Ограничение поиска по глубине поиска базы.
  - Значение по умолчанию – *sub*
  - Пример значения – *base*, *one* или *sub*
- **User Group Name Attribute** – Атрибут из записи пользователя, значения которого рассматриваются как значения группы для отправки в базу данных Access Manager. Можно указать несколько имен атрибутов, разделенных запятыми.
  - Значение по умолчанию – *memberof,ismemberof*
  - Пример значения – *memberof*, *ismemberof* или *gidNumber*
- **Enable User Search** – Параметр доступен, если выбрана опция “Enable Group Search First”.
  - Значение по умолчанию – *No*
  - Пример значения – *Yes*

5. На вкладке “Group Configs” установить свойства (Рис.2.18.):

- **Enable Group Sync** – Если для параметра “Enable Group Sync” установлено *No*, имена групп, к которым принадлежат пользователи, получены из “User Group Name Attribute”. В этом случае не применяются дополнительные групповые фильтры. Если для параметра “Enable Group Sync” установлено *Yes*, группы, к которым принадлежат пользователи, извлекаются из LDAP/AD с помощью атрибутов, связанных с группой. Включено по умолчанию, если включена функция “Incremental Sync” на вкладке “Common Configs”.
  - Значение по умолчанию – *No*
  - Пример значения – *Yes*
- **Group Member Attribute** – Имя атрибута члена группы LDAP.
  - Пример значения – *member*
- **Group Name Attribute** – Атрибут имени группы LDAP.
  - Пример значения – *distinguishedName* для AD, *cn* для OpenLdap
- **Group Object Class** – Класс объекта LDAP Group.
  - Пример значения – *group*, *groupofnames* или *posixGroup*
- **Group Search Base** – База поиска для групп. Ranger может искать несколько подразделений в AD. Модуль Ranger UserSync выполняет поиск пользователей по каждому настроенному подразделению и добавляет всех пользователей в один список. После того как все подразделения будут обработаны, членство в группе пользователей вычисляется на основе конфигурации поиска группы. Каждый сегмент подразделения должен быть разделен знаком “;” (точка с запятой).
  - Пример значения – *ou=groups,DC=example,DC=com;ou=group1;ou=group2*
- **Group Search Filter** – Дополнительный фильтр, ограничивающий группы, выбранные для синхронизации.
  - Пример значения – Для извлечения всех групп: cn=*. Для извлечения только групп, cn которых является *Engineering* или *Sales*: *(|(cn=Engineering)(cn=Sales))*
- **Enable Group Search First** – Если параметр “Enable Group Search First” не выбран: пользователи извлекаются из атрибута группы *member*. Если параметр “Enable Group Search First” выбран: членство пользователя вычисляется путем выполнения поиска LDAP на основе пользовательской конфигурации.

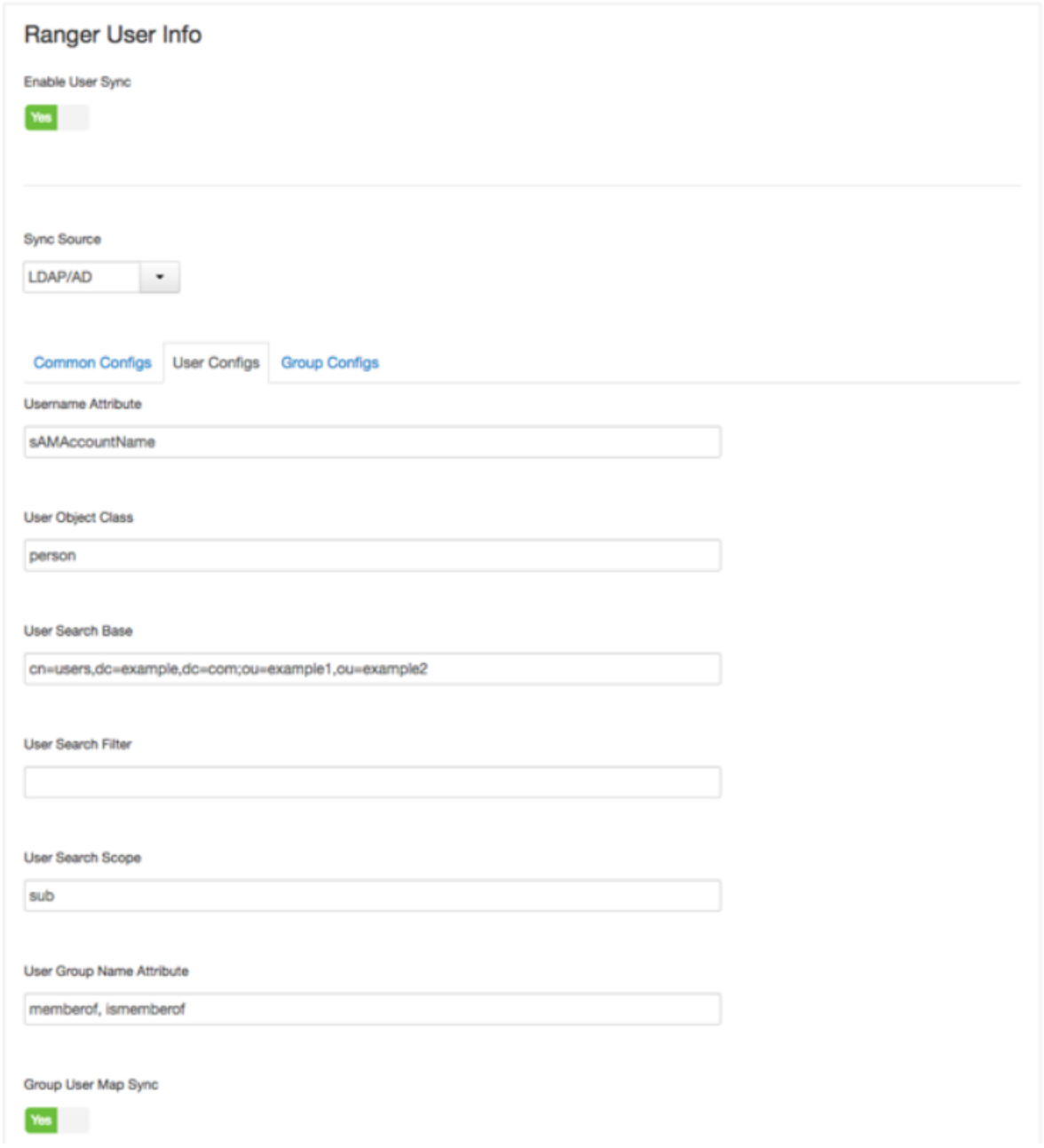


Рис.2.17.: Настройка User Configs для LDAP/AD

  - Значение по умолчанию – *No*
  - Пример значения – *Yes*

- **Sync Nested Groups** – Включает членство во вложенных группах в Ranger, чтобы права, настроенные для родительских групп, применялись ко всем членам в подгруппах. Если сама группа является членом другой группы, пользователи, принадлежащие к этой группе, также являются частью родительской группы. Уровни иерархии групп определяют глубину вложенной группы. Если свойство "Sync Nested Groups" не отображается, следует обновить Ambari 2.6.0+.
  - Значение по умолчанию – *No*
  - Пример значения – *Yes*, *No*
- **Group Hierarchy Levels** – Количество вложенных групп для оценки в поддержку "Sync Nested Groups". Задать целое число *>0*.
  - Значение по умолчанию – *0*
  - Пример значения – *2*

Автоматическое назначение роли ADMIN/KEYADMIN для внешних пользователей

Можно использовать **usersync** для пометки определенных внешних пользователей или пользователей в определенной внешней группе с ролью *ADMIN* или *KEYADMIN* в **Ranger**. Это полезно в тех случаях, когда внутренние пользователи не могут войти в **Ranger**.

1. В "Ambari>Ranger>Configs>Advanced>Custom ranger-ugsync-site" выбрать "Add Property";
2. Добавить следующие свойства:

- *ranger.usersync.role.assignment.list.delimiter* = **&**
  - Значение по умолчанию – "&"
- *ranger.usersync.users.groups.assignment.list.delimiter* = **:**
  - Значение по умолчанию – ":"
- *ranger.usersync.username.groupname.assignment.list.delimiter* = **,**
  - Значение по умолчанию – ","
  - *ranger.usersync.group.based.role.assignment.rules* =

```
ROLE_SYS_ADMIN:u:userName1,userName2&ROLE_SYS_ADMIN:g:groupName1,groupName2&ROLE_KEY_ADMIN:u:userName&
↪ROLE_KEY_ADMIN:g:groupName&ROLE_USER:u:userName3,userName4&ROLE_USER:g:groupName
```

3. Нажать *Add*;
4. Перезапустить Ranger.

   Пример:

```
ranger.usersync.role.assignment.list.delimiter = &
ranger.usersync.users.groups.assignment.list.delimiter = :
ranger.usersync.username.groupname.assignment.list.delimiter = ,
ranger.usersync.group.based.role.assignment.rules : &ROLE_SYS_ADMIN:u:ldapuser_12,ldapuser2
```

Ranger Tagsync

Для настройки **Ranger Tagsync** следует на странице "Customize Services" на вкладке "Ranger Tagsync" выбрать необходимый **Tag Source** путем проставления флага в соответствующее поле (Рис.2.19.):

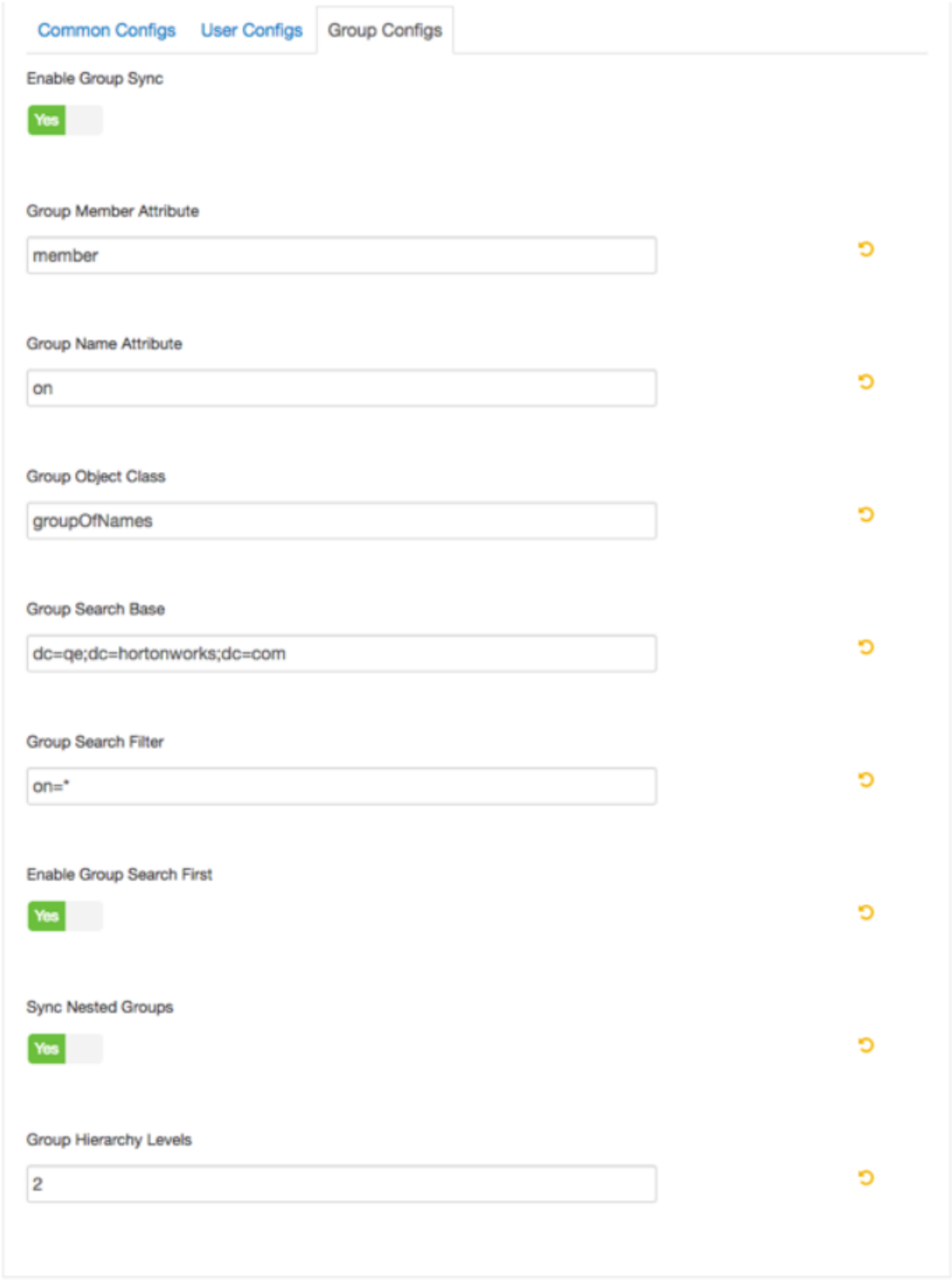


Рис.2.18.: Настройка Group Configs для LDAP/AD

- Atlas Tag Source;
- AtlasREST Tag Source;
- File Tag Source.

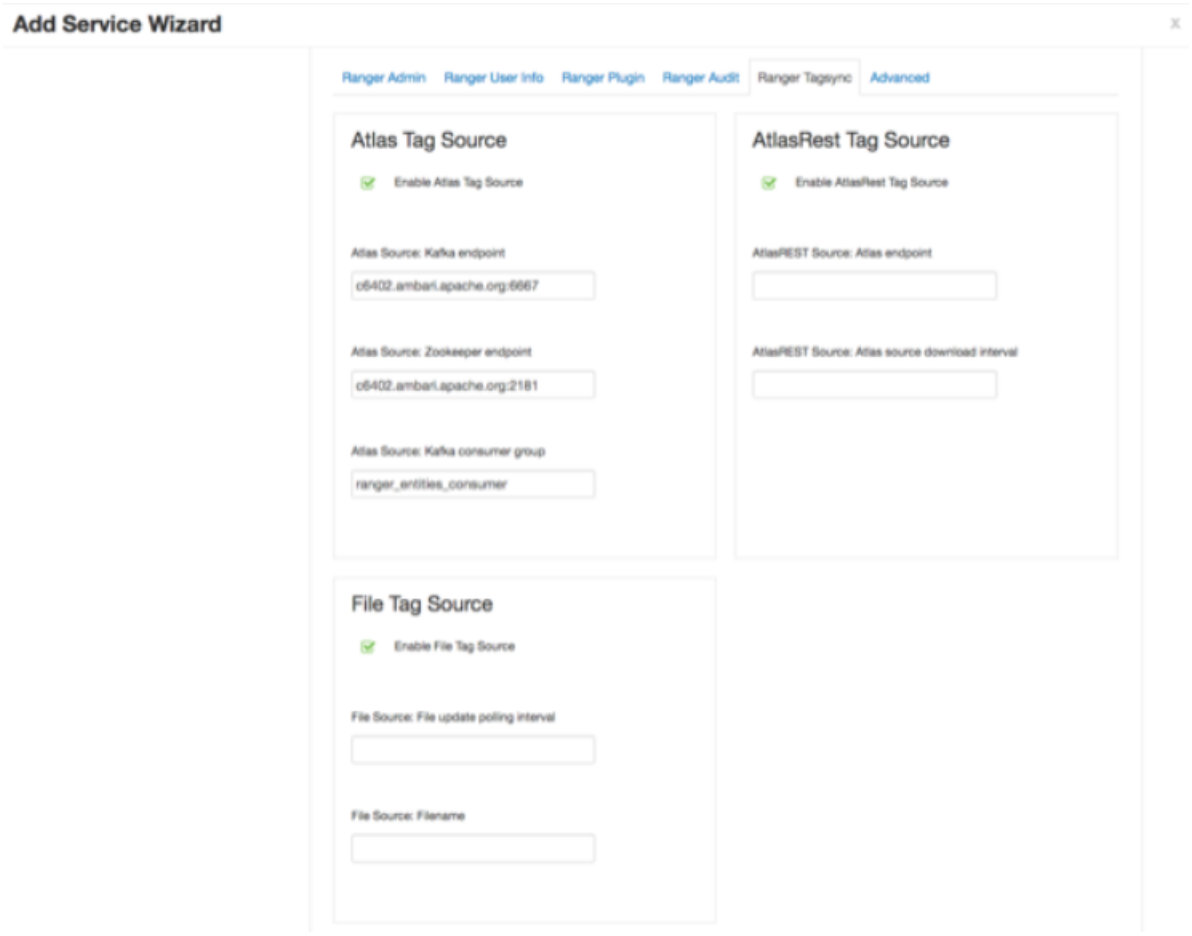


Рис.2.19.: Ranger Tagsync

Описание свойств **Tag Source** приведено в таблицах.

Таблица2.6.: Atlas Tag Source

Свойство	Описание
Atlas Source: Kafka endpoint	Конечная точка Kafka: *<kafka_server_url>:6667*
Atlas Source: ZooKeeper endpoint	Конечная точка ZooKeeper: *<zookeeper_server_url>*:2181
Atlas Source: Kafka consumer group	Пользователь Ranger

Таблица2.7.: AtlasREST Tag Source

Свойство	Описание
AtlasREST Source: Atlas endpoint	Конечная точка AtlasREST: *<atlas_host_url>:21000*
AtlasREST Source: Atlas source download interval	Интервал загрузки AtlasREST (миллисекунды)

Таблица2.8.: File Tag Source

Свойство	Описание
File Source: File update polling interval	Интервал опроса обновлений файла (миллисекунды)
File Source: Filename	Имя файла tag source

Ranger Authentication

В разделе описывается, как настроить аутентификацию **Ranger** для **UNIX**, **LDAP** и **AD**:

- *Ranger UNIX Authentication*
- *Ranger LDAP Authentication*
- *Ranger Active Directory Authentication*

После завершения настройки параметров аутентификации нажать кнопку *Next* для продолжения установки. Затем обновить конфигурацию **Ranger admin truststore**, добавив следующие параметры в "Ambari> Ranger> Configs> Advanced> Advanced ranger-admin-site":

```
ranger.truststore.file=/etc/ranger/admin/truststore
ranger.truststore.password=password
```

И перезапустить Ranger.

Ranger UNIX Authentication

Для настройки аутентификации **Ranger** для **UNIX** необходимо выполнить следующий порядок действий:

1. Перейти на вкладку "Advanced" на странице "Customize Services" (см. Рис.2.7.);
2. На открывшейся странице в разделе "Ranger Settings" указать адрес хоста Ranger Access Manager/Service Manager в поле "External URL" в формате *http://<your_ranger_host>:6080* (Рис.2.20.);
3. В поле "Authentication method" отметить *UNIX*. *HTTP* включен по умолчанию – если отключить *HTTP*, то возможен только *HTTPS*;
4. В блоке "UNIX Authentication Settings" указать свойства:

- **Allow remote Login** – Флаг для включения/отключения удаленного входа.
  - Значение по умолчанию – *true*
  - Пример значения – *true*
- **ranger.unixauth.service.hostname** – Адрес хоста, на котором запущена служба проверки подлинности UNIX.
  - Значение по умолчанию – *{{ugsync_host}}*
  - Пример значения – *{{ugsync_host}}*
- **ranger.unixauth.service.port** – Номер порта, на котором запущена служба проверки подлинности UNIX.
  - Значение по умолчанию – *5151*
  - Пример значения – *5151*

Свойства со значением {{xyz}} – это макропеременные, которые производятся из других заданных значений, для оптимизации процесса настройки. Переменные доступны для редактирования. Для восстановления исходного значения следует нажать значок *Set Recommended* справа от поля свойства.

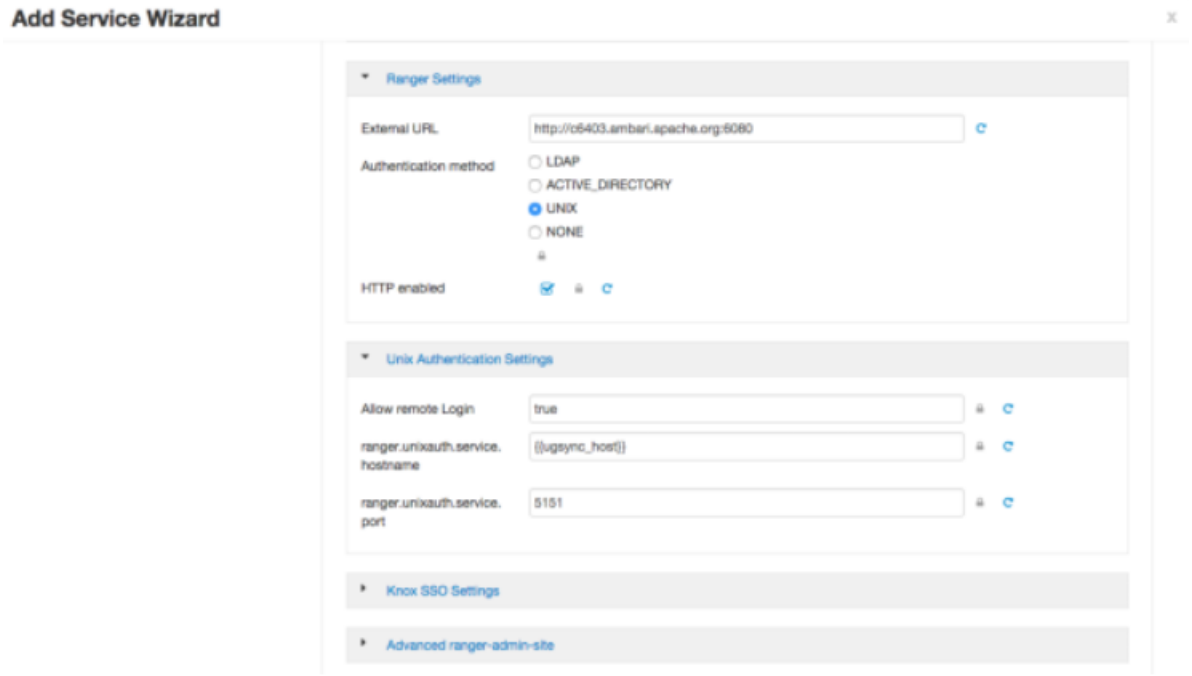


Рис.2.20.: Настройка Ranger UNIX Authentication

Ranger LDAP Authentication

Для настройки аутентификации **Ranger** для **LDAP** необходимо выполнить следующий порядок действий:

1. Перейти на вкладку “Advanced” на странице “Customize Services” (см. Рис.2.7.);
2. На открывшейся странице в разделе “Ranger Settings” указать адрес хоста Ranger Access Manager/Service Manager в поле “External URL” в формате *http://<your_ranger_host>:6080* (Рис.2.21.);
3. В поле “Authentication method” отметить *LDAP*;
4. В блоке “LDAP Settings” указать свойства:

- **ranger.ldap.base.dn** – Distinguished Name (DN) начальной точки для поиска на сервере каталогов.
  - Значение по умолчанию – *dc=example,dc=com*
  - Пример значения – *dc=example,dc=com*
- **Bind User** – Полное Distinguished Name (DN), включая Common Name (CN) учетной записи пользователя LDAP с правами поиска пользователей. Это значение макропеременной, полученное из значения “Bind User” из “Ranger User Info > Common Configs”.
  - Значение по умолчанию – *{{ranger_ug_ldap_bind_dn}}*
  - Пример значения – *{{ranger_ug_ldap_bind_dn}}*
- **Bind User Password** – Пароль для Bind User. Это значение макропеременной, которое получено из значения пароля “Bind User” из “Ranger User Info > Common Configs”.
- **ranger.ldap.group. roleattribute** – Атрибут роли группы LDAP.
  - Значение по умолчанию – *cn*
  - Пример значения – *cn*
- **ranger.ldap.referral** – Существует три возможных значения:

  - *follow* – сервис LDAP сначала обрабатывает все обычные записи, а затем следует по ссылкам;
  - *throw* – все нормальные записи возвращаются в перечислении до того, как выбрано *ReferralException*. При этом в случаях настройки свойства на *follow* или *throw* ответ об ошибке "referral" обрабатывается немедленно;
  - *ignore* – указывает, что сервер должен возвращать записи ссылок как обычные записи, обычный текст. Это может привести к частичным результатам поиска.

  Рекомендуемая настройка *follow*. При поиске в каталоге сервер может возвращать несколько результатов поиска, а также несколько ссылок, которые показывают, где получить дальнейшие результаты. Эти результаты и ссылки могут чередоваться на уровне протокола.

  - Значение по умолчанию – *ignore*
  - Пример значения – *follow | ignore | throw*

- **LDAP URL** – URL-адрес сервера LDAP. Это значение макропеременной, полученное из значения "LDAP/AD URL" из "Ranger User Info > Common Configs".
  - Значение по умолчанию – *{{ranger_ug_ldap_url}}*
  - Пример значения – *{{ranger_ug_ldap_url}}*
- **ranger.ldap.user. dnpattern** – Шаблон DN пользователя расширяется при входе пользователя в систему. Например, если пользователь *ldapadmin* выполняет вход, сервер LDAP попытается связаться с DN *uid=ldapadmin,ou=users,dc=example,dc=com*, используя пароль, предоставленный пользователем.
  - Значение по умолчанию – *uid={0},ou=users,dc=xasecure,dc=net*
  - Пример значения – *cn=ldapadmin,ou=Users,dc=example,dc=com*
- **User Search Filter** – Фильтр поиска, используемый для Bind Authentication. Это значение макропеременной, полученное из значения "User Search Filter" из "Ranger User Info > Common Configs".
  - Значение по умолчанию – *{{ranger_ug_ldap_user_searchfilter}}*
  - Пример значения – *{{ranger_ug_ldap_user_searchfilter}}*

Свойства со значением *{{xyz}}* – это макропеременные, которые производятся из других заданных значений, для оптимизации процесса настройки. Переменные доступны для редактирования. Для восстановления исходного значения следует нажать значок *Set Recommended* справа от поля свойства.

Ranger Active Directory Authentication

Для настройки аутентификации **Ranger** для **Active Directory** необходимо выполнить следующий порядок действий:

1. Перейти на вкладку "Advanced" на странице "Customize Services" (см. Рис.2.7.);
2. На открывшейся странице в разделе "Ranger Settings" указать адрес хоста Ranger Access Manager/Service Manager в поле "External URL" в формате *http://<your_ranger_host>:6080* (Рис.2.22.);
3. В поле "Authentication method" отметить *ACTIVE_DIRECTORY*;
4. В блоке "AD Settings" указать свойства:

- **ranger.ldap.ad.base.dn** – Distinguished Name (DN) начальной точки для поиска на сервере каталогов.
  - Значение по умолчанию – *dc=example,dc=com*
  - Пример значения – *dc=example,dc=com*

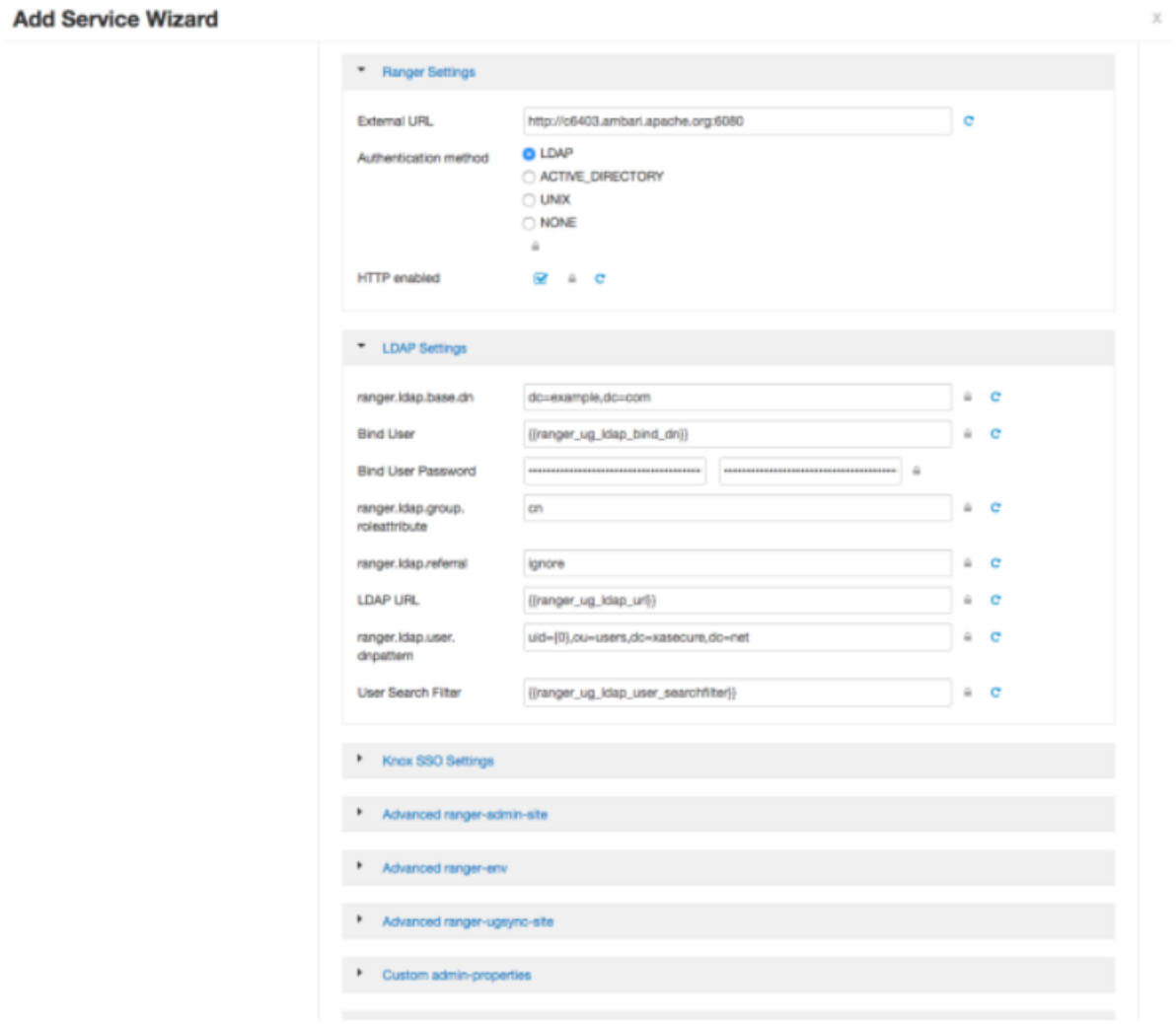


Рис.2.21.: Настройка Ranger LDAP Authentication

- **ranger.ldap.ad.bind.dn** – Полное Distinguished Name (DN), включая Common Name (CN) учетной записи пользователя LDAP с правами поиска пользователей. Это значение макропеременной, полученное из значения “Bind User” из “Ranger User Info > Common Configs”.
  - Значение по умолчанию – *{{ranger_ug_ldap_bind_dn}}*
  - Пример значения – *{{ranger_ug_ldap_bind_dn}}*
- **ranger.ldap.ad.bind.password** – Пароль для bind.dn. Это значение макропеременной, полученное из значения “Bind User Password” из “Ranger User Info > Common Configs”.
- **Domain Name (Only for AD)** – Доменное имя сервера аутентификации AD
  - Пример значения – *dc=example,dc=com*
- **ranger.ldap.ad.referral** – Существует три возможных значения:
  - *follow* – сервис LDAP сначала обрабатывает все обычные записи, а затем следует по ссылкам;
  - *throw* – все нормальные записи возвращаются в перечислении до того, как выбрано *ReferralException*. При этом в случаях настройки свойства на *follow* или *throw* ответ об ошибке “referral” обрабатывается немедленно;
  - *ignore* – указывает, что сервер должен возвращать записи ссылок как обычные записи, обычный текст. Это может привести к частичным результатам поиска.

  Рекомендуемая настройка *follow*. При поиске в каталоге сервер может возвращать несколько результатов поиска, а также несколько ссылок, которые показывают, где получить дальнейшие результаты. Эти результаты и ссылки могут чередоваться на уровне протокола.
  - Значение по умолчанию – *ignore*
  - Пример значения – *follow | ignore | throw*
- **ranger.ldap.ad.url** – URL-адрес сервера AD. Это значение макропеременной, полученное из значения “LDAP/AD URL” из “Ranger User Info > Common Configs”.
  - Значение по умолчанию – *{{ranger_ug_ldap_url}}*
  - Пример значения – *{{ranger_ug_ldap_url}}*
- **ranger.ldap.ad.user.searchfilter** – Фильтр поиска, используемый для Bind Authentication. Это значение макропеременной, полученное из значения “User Search Filter” из “Ranger User Info > Common Configs”.
  - Значение по умолчанию – *{{ranger_ug_ldap_user_searchfilter}}*
  - Пример значения – *{{ranger_ug_ldap_user_searchfilter}}*

Свойства со значением *{{xyz}}* – это макропеременные, которые производятся из других заданных значений, для оптимизации процесса настройки. Переменные доступны для редактирования. Для восстановления исходного значения следует нажать значок *Set Recommended* справа от поля свойства.

5. При сохранении метода проверки подлинности Active Directory может появиться всплывающее окно “Dependent Configurations”, рекомендующее установить метод проверки подлинности LDAP. Эта рекомендуемая конфигурация не должна применяться для AD, поэтому необходимо очистить (отменить) параметр *ranger.authentication.method*, а затем нажать кнопку *OK* (Рис.2.23.).

2.2.3 Завершение установки

Завершение процесса установки **Ranger** осуществляется в 3 шага:

1. На странице “Review” внимательно проверить заданные параметры конфигурации. Затем для установки Ranger на сервер Ambari нажать кнопку *Deploy* (Рис.2.24.).

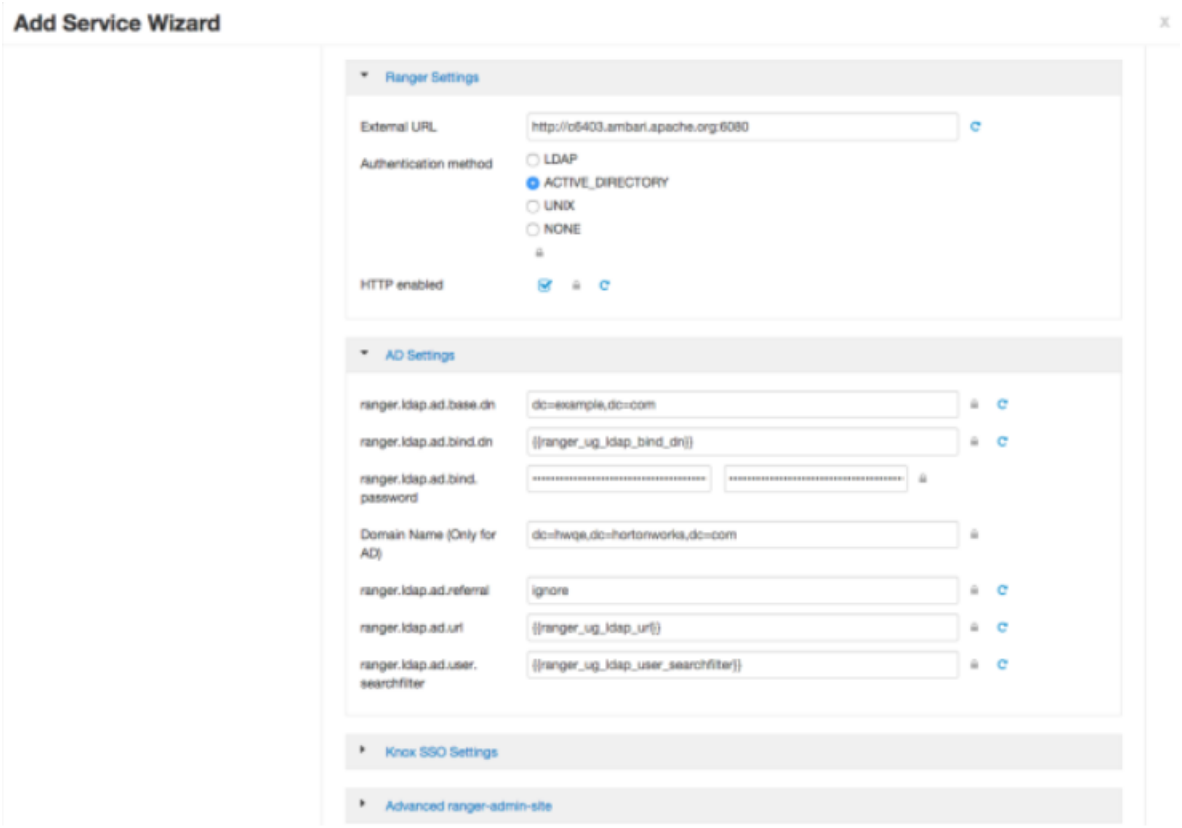


Рис.2.22.: Настройка Ranger Active Directory Authentication

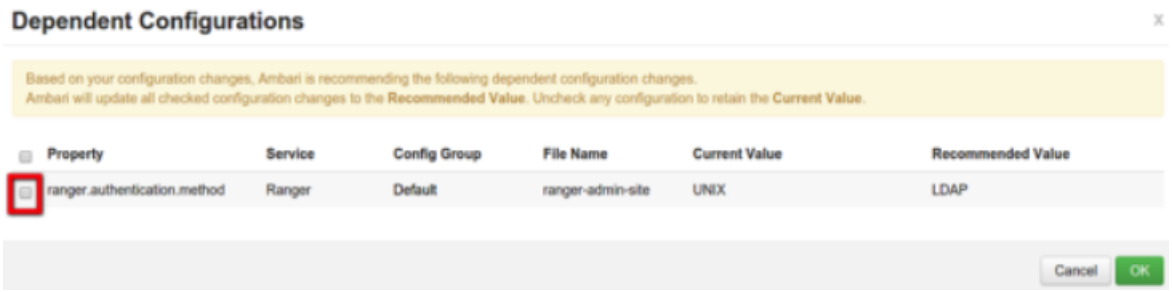


Рис.2.23.: Dependent Configurations

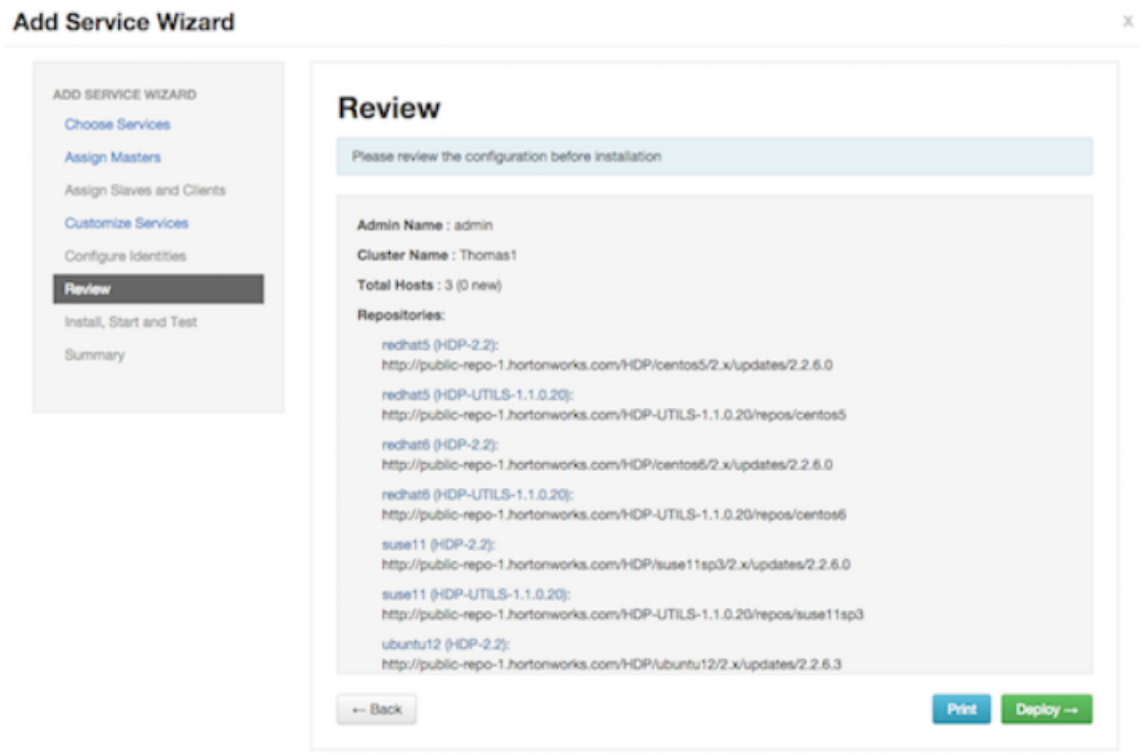


Рис.2.24.: Проверка установленных параметров конфигурации

2. Ranger устанавливается на указанном хосте на сервере Ambari. Индикатор выполнения отображает ход установки (Рис.2.25.).
3. По завершении установки на странице “Summary” отображаются детали установки. Может потребоваться перезапуск служб для компонентов кластера.

Important: В случае сбоя установки необходимо завершить процесс установки, а затем перенастроить и переустановить Ranger

2.2.4 Расширенные настройки пользователей

Для получения доступа к расширенным настройкам пользователя необходимо выбрать вкладку “Advanced” на странице “Customize Service”. **Usersync** загружает пользователей из **UNIX**, **LDAP** или **AD** и заполняет ими локальные таблицы пользователей **Ranger**.

- *Настройки UNIX Usersync*
- *Необходимые настройки LDAP и AD Usersync*
- *Дополнительные настройки LDAP и AD Usersync*

Important: Чтобы гарантировать, что авторизация уровня LDAP/AD применяется в Hadoop, следует сначала настроить Hadoop Group Mapping для LDAP/AD: *Настройка сопоставления групп Hadoop для LDAP/AD*

Important: Перед применением изменений рекомендуется протестировать Usersync, чтобы пользователи и



Рис.2.25.: Отображение хода установки

группы извлекались по назначению: *Тест-драйв Ranger Usersync*

После указания всех настроек на странице “Customize Services” следует нажать кнопку *Next* для продолжения установки.

Настройки UNIX Usersync

При использовании аутентификации **UNIX** значения по умолчанию для свойств *Advanced ranger-ugsync-site* – это настройки для проверки подлинности **UNIX** (Рис.2.26.).

Необходимые настройки LDAP и AD Usersync

При использовании аутентификации **LDAP** необходимо обновить следующие свойства *Advanced ranger-ugsync-site*:

Таблица2.9.: Настройки LDAP Advanced ranger-ugsync-site

Свойство	Значение LDAP
ranger.usersync.ldap.bindkeystore	Установить значение таким же, как и в свойстве *ranger.usersync.credstore.filename*. Значение по умолчанию: /usr/hdp/current/ranger-usersync/conf/ugsync.jceks
ranger.usersync.ldap.bindalias	ranger.usersync.ldap.bindalias
ranger.usersync.source.impl.class	ldap

Таблица2.10.: Настройки AD Advanced ranger-ugsync-site

Свойство	Значение AD
ranger.usersync.source.impl.class	ldap

Дополнительные настройки LDAP и AD Usersync

При использовании проверки подлинности **LDAP** или **Active Directory** может потребоваться обновление свойств в зависимости от конкретных характеристик развертывания:

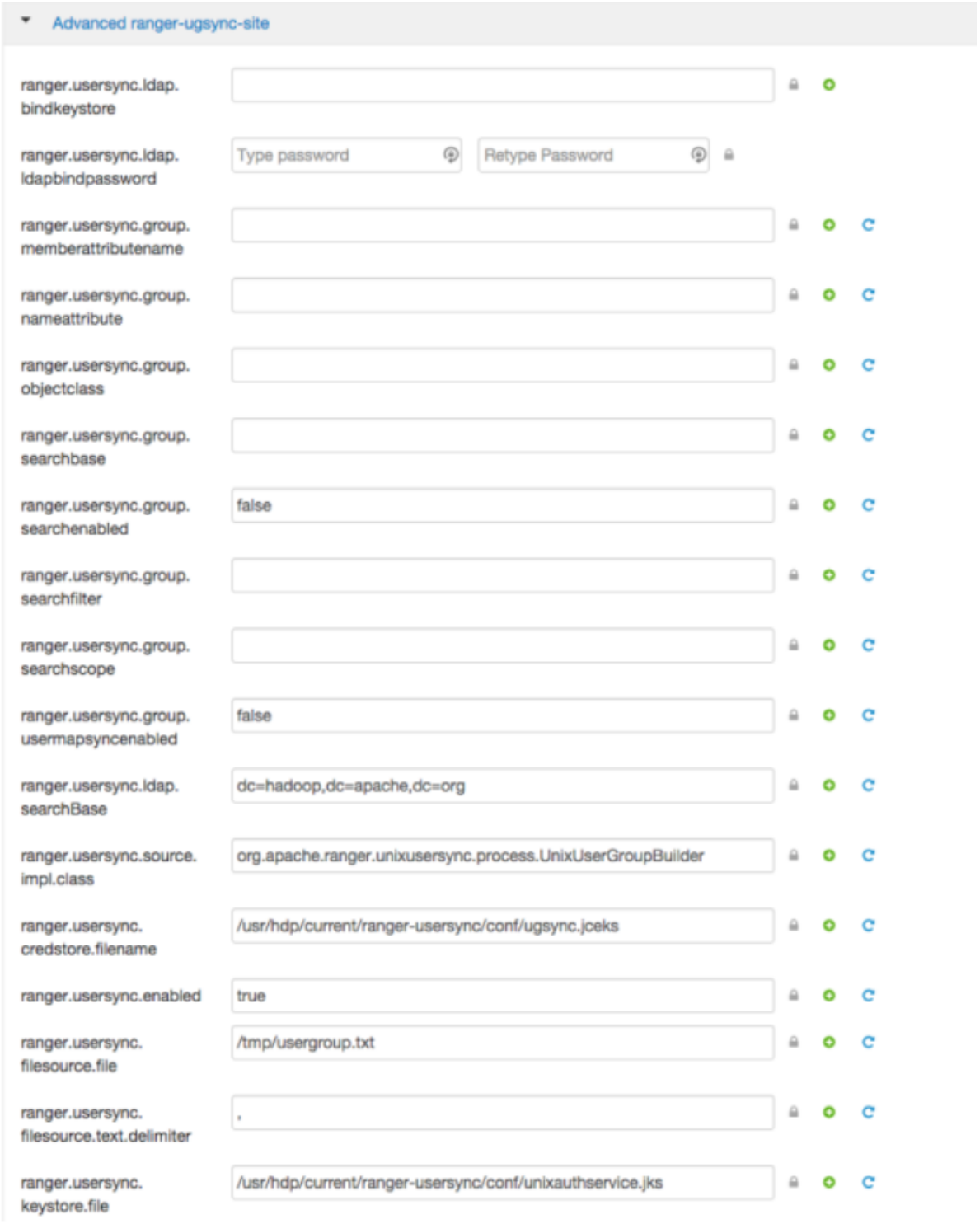


Рис.2.26.: Свойства Advanced ranger-ugsync-site

- **ranger.usersync.ldap.url**
  - Значение LDAP: *ldap://127.0.0.1:389*
  - Значение AD: *ldap://ad-conrowoller-hostname:389*
- **ranger.usersync.ldap.binddn**
  - Значение LDAP: *cn=ldapadmin,ou=users, dc=example,dc=com*
  - Значение AD: *cn=adadmin,cn=Users, dc=example,dc=com*
- **ranger.usersync.ldap.ldapbindpassword**
  - Значение LDAP: *secret*
  - Значение AD: *secret*
- **ranger.usersync.ldap.searchBase**
  - Значение LDAP: *dc=example,dc=com*
  - Значение AD: *dc=example,dc=com*
- **ranger.usersync.source.impl.class**
  - Значение LDAP: *org.apache.ranger. ladpusersync. process.LdapUserGroupBuilder*
- **ranger.usersync.ldap.user.searchbase**
  - Значение LDAP: *ou=users, dc=example, dc=com*
  - Значение AD: *dc=example,dc=com*
- **ranger.usersync.ldap.user.searchscope**
  - Значение LDAP: *sub*
  - Значение AD: *sub*
- **ranger.usersync.ldap.user.objectclass**
  - Значение LDAP: *person*
  - Значение AD: *person*
- **ranger.usersync.ldap.user.searchfilter**
  - Значение LDAP: *Set to single empty space if no value. Do not leave it as "empty"*
  - Значение AD: *(objectcategory=person)*
- **ranger.usersync.ldap.user.nameattribute**
  - Значение LDAP: *uid or cn*
  - Значение AD: *sAMAccountName*
- **ranger.usersync.ldap.user.groupnameattribute**
  - Значение LDAP: *memberof,ismemberof*
  - Значение AD: *memberof,ismemberof*
- **ranger.usersync.ldap.username.caseconversion**
  - Значение LDAP: *none*
  - Значение AD: *none*
- **ranger.usersync.ldap.groupname.caseconversion**

  - Значение LDAP: *none*
  - Значение AD: *none*

Следующие свойства применяются при фильтровке групп:

- **ranger.usersync.group.searchenabled**
  - Значение LDAP: *false*
  - Значение AD: *false*
- **ranger.usersync.group.usermapsyncenabled**
  - Значение LDAP: *false*
  - Значение AD: *false*
- **ranger.usersync.group.searchbase**
  - Значение LDAP: *ou=groups, dc=example, dc=com*
  - Значение AD: *dc=example,dc=com*
- **ranger.usersync.group.searchscope**
  - Значение LDAP: *sub*
  - Значение AD: *sub*
- **ranger.usersync.group.objectclass**
  - Значение LDAP: *groupofnames*
  - Значение AD: *groupofnames*
- **ranger.usersync.group.searchfilter**
  - Значение LDAP: *needed for AD authentication*
  - Значение AD: *(member=CN={0}, OU=MyUsers, DC=AD-HDP, DC=COM)*
- **ranger.usersync.group.nameattribute**
  - Значение LDAP: *cn*
  - Значение AD: *cn*
- **ranger.usersync.group.memberattributename**
  - Значение LDAP: *member*
  - Значение AD: *member*
- **ranger.usersync.pagedresultsenabled**
  - Значение LDAP: *true*
  - Значение AD: *true*
- **ranger.usersync.pagedresultssize**
  - Значение LDAP: *500*
  - Значение AD: *500*
- **ranger.usersync.user.searchenabled**
  - Значение LDAP: *false*
  - Значение AD: *false*

- **ranger.usersync.group.search.first.enabled**
  - Значение LDAP: *false*
  - Значение AD: *false*

2.2.5 Настройка Ranger для LDAP SSL

Можно использовать следующие настройки **LDAP SSL** с помощью самоподписанных сертификатов в стандартном **Ranger User Sync TrustStore**:

1. Для свойства *ranger.usersync.truststore.file* расположение по умолчанию */usr/hdp/current/ranger-usersync/conf/mytruststore.jks*;
2. Скопировать и отредактировать самоподписанные сертификаты;
3. Установить свойство *ranger.usersync.truststore.file* в новый файл:

```
cd /usr/hdp/<version>/ranger-usersync
service ranger-usersync stop
service ranger-usersync start
```

Сертификат LDAPS содержится в *cert.pem*.

2.2.6 Настройка пользователей без использования учетных данных DBA

С целью не предоставления деталей учетной записи администратора базы данных (DBA) установщику **Ambari Ranger** можно использовать скрипт **Python** *dba_script.py* для создания пользователей базы данных **Ranger DB** без передачи информации об учетной записи DBA. После этого можно запустить обычную установку **Ambari Ranger** без указания имени и пароля администратора.

Создание пользователей **Ranger DB** при помощи скрипта *dba_script.py*:

1. Загрузить Ranger rpm с помощью команды *yum install*:

```
yum install ranger-admin
```

2. В каталоге */usr/hdp/current/ranger-admin* должен быть файл с именем *dba_script.py*;
3. Получить внутренний скрипт и убедиться, что DBA имеет право запускать его;
4. Выполнить скрипт командой:

```
python dba_script.py
```

5. Указать все необходимые значения в аргументе (включает *db flavor*, *JDBC jar*, *db host*, *db name*, *db user* и другие параметры):

- Если во время выполнения не предпочитается передача аргументов в командной строке, можно обновить файл */usr/hdp/current/ranger-admin/install.properties*, а затем выполнить команду:

```
python dba_script.py -q
```

При указании опции *-q* скрипт считывает всю необходимую информацию из файла *install.properties*;

- Опция *-d* используется для запуска скрипта в режиме “dry”. Это приводит к созданию сценария базы данных:

```
python dba_script.py -d /tmp/generated-script.sql
```

Сценарий может выполнить любой пользователь, но рекомендуется, чтобы его запустил в режиме "dry" системный администратор баз данных. В любом случае системный DBA должен просматривать сгенерированный скрипт, но при этом вносить лишь незначительные корректировки, например, изменение расположения конкретного файла базы данных. Не следует вносить существенных изменений, которые могут сильно изменить скрипт – в противном случае установка Ranger может завершиться ошибкой.

Затем системному администратору баз данных необходимо запустить созданный скрипт.

6. Запустить процедуру установки Ranger Ambari, предварительно установив на странице "Customize Services" в разделе "Ranger Admin" для параметра *Setup Database and Database User* значение *No*.

2.2.7 Обновление паролей администратора Ranger

При обновлении паролей на странице "Ranger Configs" для нижеприведенных пользователей необходимо также обновить пароли каждого компонента **Ambari**, для которого включен плагин **Ranger**.

Important: Индивидуальные конфигурации компонентов Ambari не обновляются автоматически – перезапуск сервиса завершается ошибкой, если пароли для каждого компонента не обновлены

- Ranger Admin user – учетные данные пользователя устанавливаются в "Configs > Advanced ranger-env" в полях "admin_username" (значение по умолчанию: *admin*) и "admin_password" (значение по умолчанию: *admin*);
- Admin user, используемый Ambari для создания репозитория/политик – имя пользователя задается в "Configs > Admin Settings" в поле "Ranger Admin username for Ambari" (значение по умолчанию: *amb_ranger_admin*). Пароль устанавливается в поле "Ranger Admin user's password for Ambari" (задается во время установки Ranger).

На рисунке показано расположение полей с перечисленными параметрами на странице настроек "Ranger Configs" (Рис.2.27.).

2.2.8 Включение плагинов Ranger

Плагины **Ranger** могут быть включены для нескольких сервисов **ADH**. По соображениям производительности рекомендуется хранить аудиты в **Solr** и **HDFS**, а не в базе данных.

При использовании кластера с поддержкой **Kerberos** необходимо выполнить ряд дополнительных шагов, чтобы убедиться в возможности использования подключаемых плагинов **Ranger** в кластере **Kerberos** (*HDFS в кластере с поддержкой Kerberos*).

Доступны следующие плагины **Ranger**: *HDFS*, **Hive**, **HBase**, **Kafka**, **Knox**, **YARN**, **Storm**, **Atlas**.

HDFS

Для включения плагина **Ranger HDFS** необходимо выполнить следующие действия:

1. На странице "Ranger Configs" выбрать вкладку "Ranger Plugin" (Рис.2.28.).
2. В поле "HDFS Ranger Plugin" активировать кнопку *On* и сохранить действие.
3. При этом появляется всплывающее окно "Save Configuration". Необходимо ввести примечание с описанием только что внесенных изменений и сохранить кнопкой *Save* (Рис.2.29.).
4. При этом появляется всплывающее окно "Dependent Configuration". Для подтверждения обновлений конфигурации необходимо нажать кнопку *OK* (Рис.2.30.).
5. Нажать кнопку *OK* во всплывающем окне сохранения настроек "Save Configuration Changes" (Рис.2.31.).
6. Перейти в меню навигации на пункт "HDFS", затем выбрать "Restart > Restart All Affected" для перезапуска сервиса HDFS и загрузки новой конфигурации (Рис.2.32.).

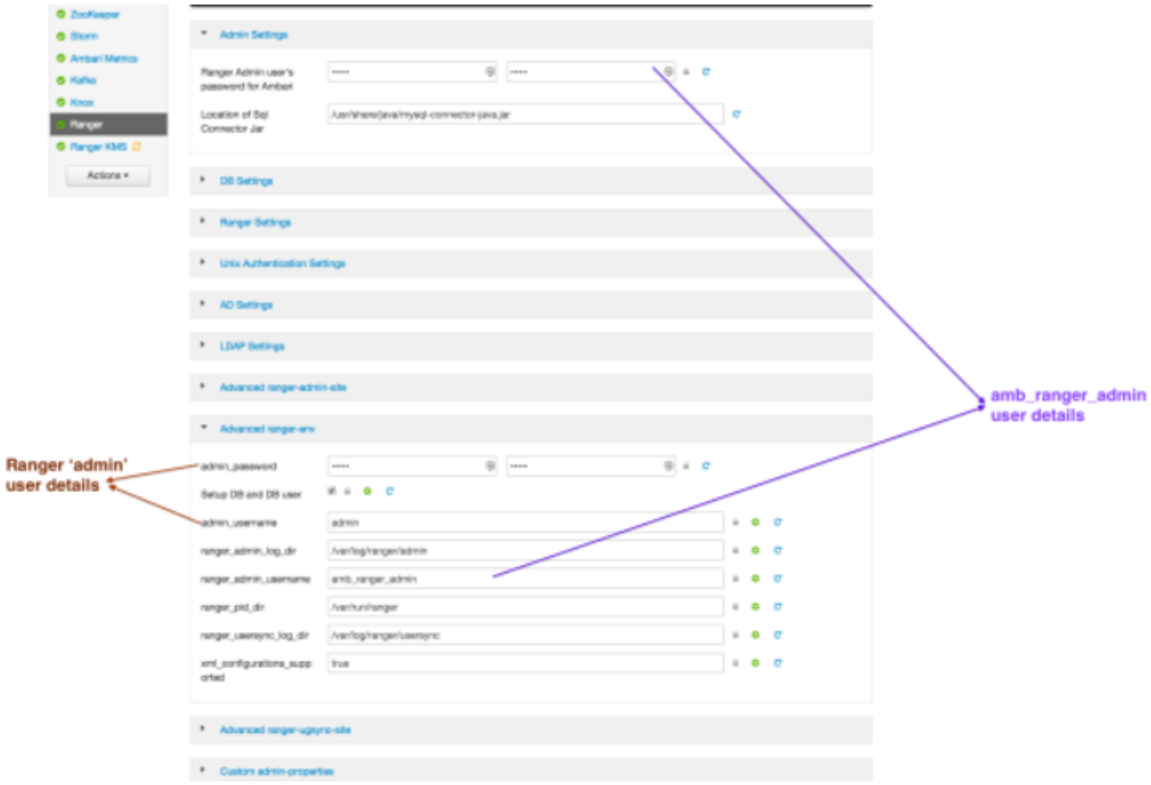


Рис.2.27.: Обновление паролей администратора Ranger

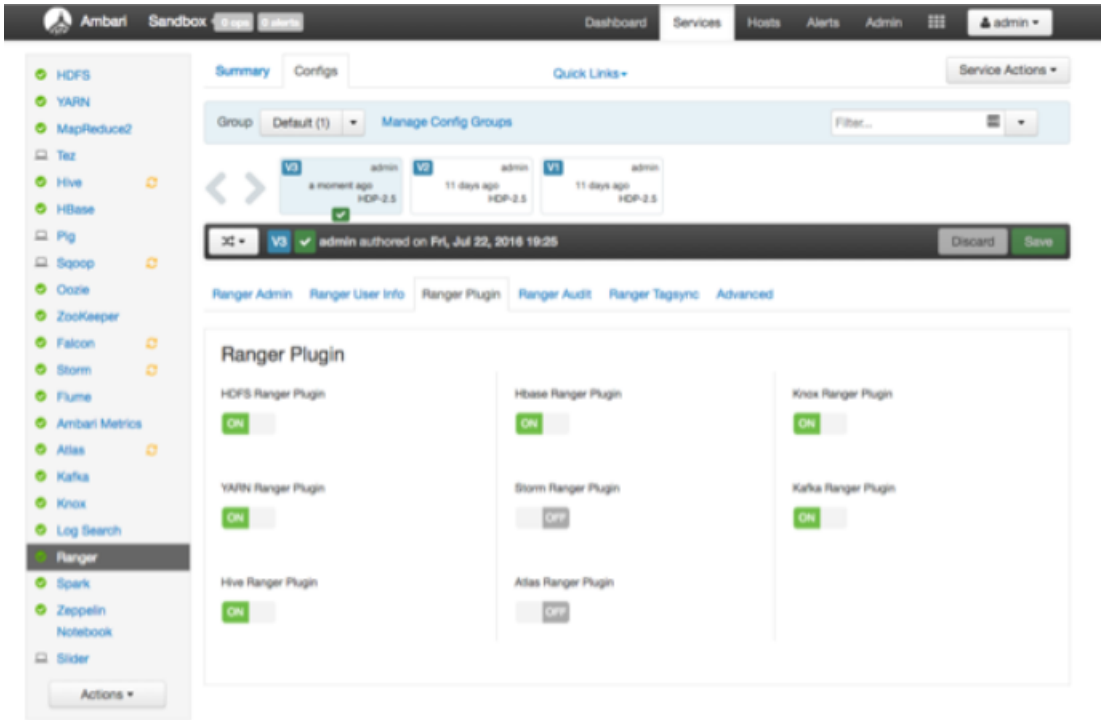


Рис.2.28.: Ranger Plugin

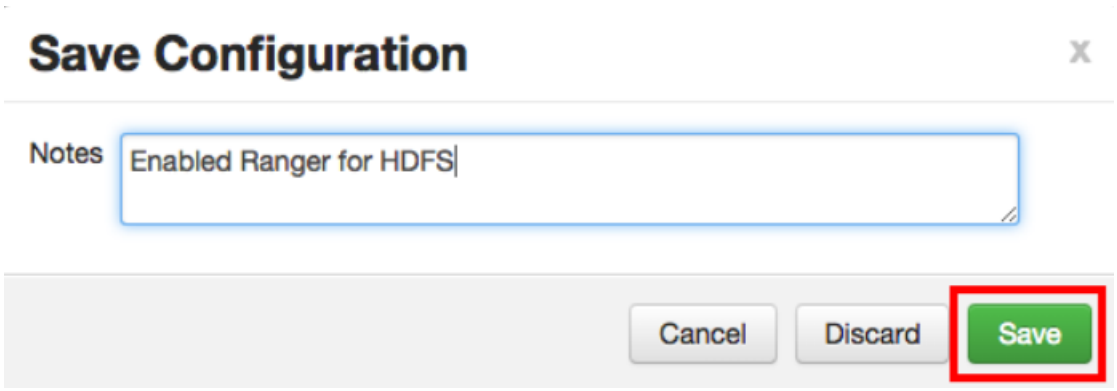


Рис.2.29.: Save Configuration

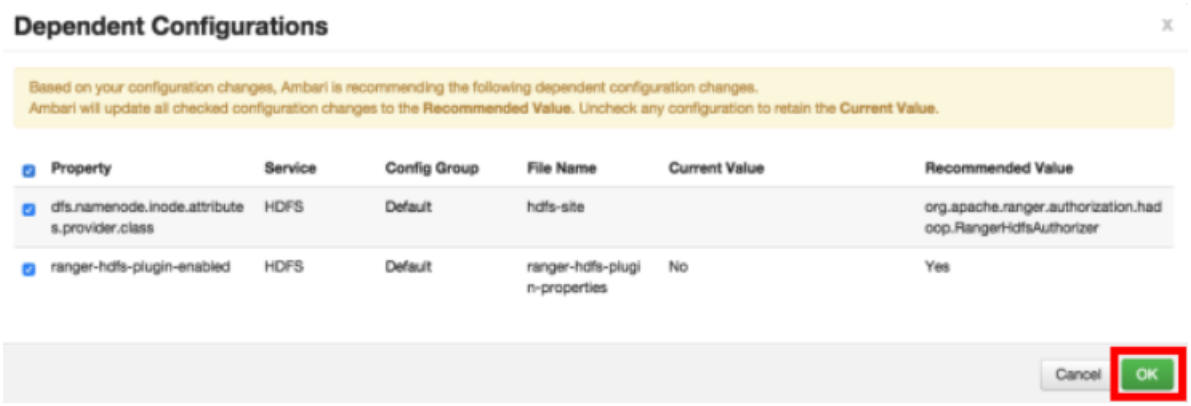


Рис.2.30.: Dependent Configuration

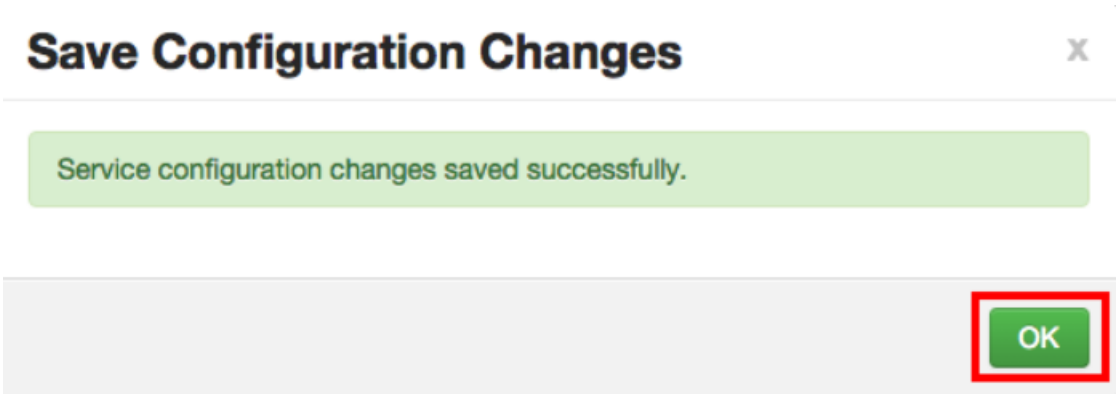


Рис.2.31.: Save Configuration Changes

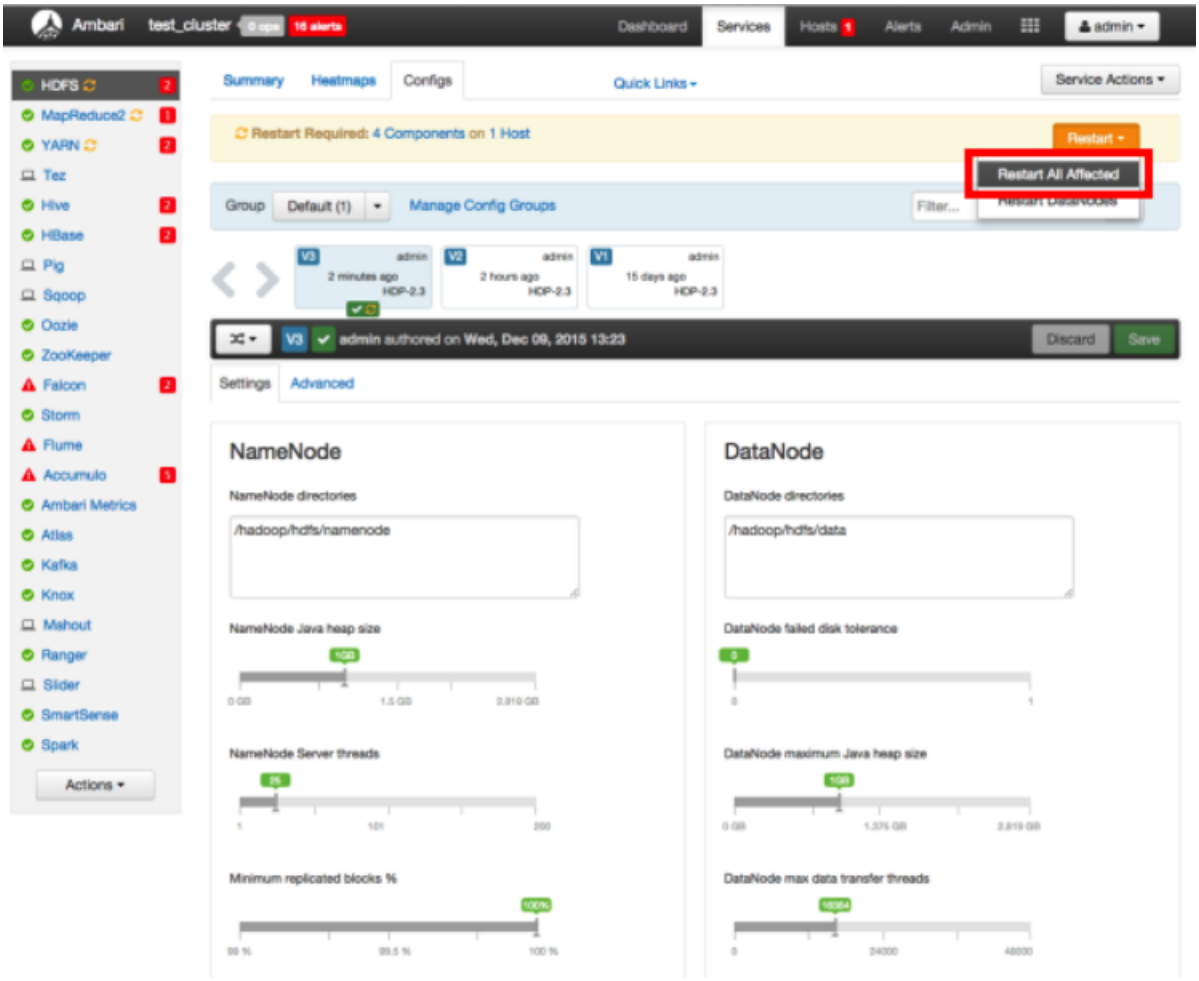


Рис.2.32.: Restart All Affected

7. Нажать *Confirm Restart All* во всплывающем окне “Confirmation” для подтверждения перезапуска HDFS (Рис.2.33.).
8. После перезапуска HDFS плагин Ranger для HDFS будет включен. Другие компоненты могут также потребовать перезагрузки.

HDFS в кластере с поддержкой Kerberos

Для включения плагина **Ranger HDFS** в кластере с поддержкой **Kerberos** необходимо выполнить следующие действия:

1. Создать пользователя системы *rangerhdfslookup*. Убедиться, что пользователь синхронизирован с *Ranger Admin* (на вкладке “Settings > Users/Groups” в интерфейсе “Ranger Admin User Interface”);
2. Создать принципала Kerberos для *rangerhdfslookup*, введя следующую команду (один пользователь/принципал, например, *rangerrepouser*, может быть создан и использован в разных сервисах):

```
kadmin.local -q 'addprinc -pw rangerhdfslookup rangerhdfslookup@example.com
```

3. Перейти в разделе сервиса “HDFS” на вкладку “Config”;
4. В блоке “Advanced ranger-hdfs-plugin-properties” обновить свойства, перечисленные в таблице под рисунком (Рис.2.34.).

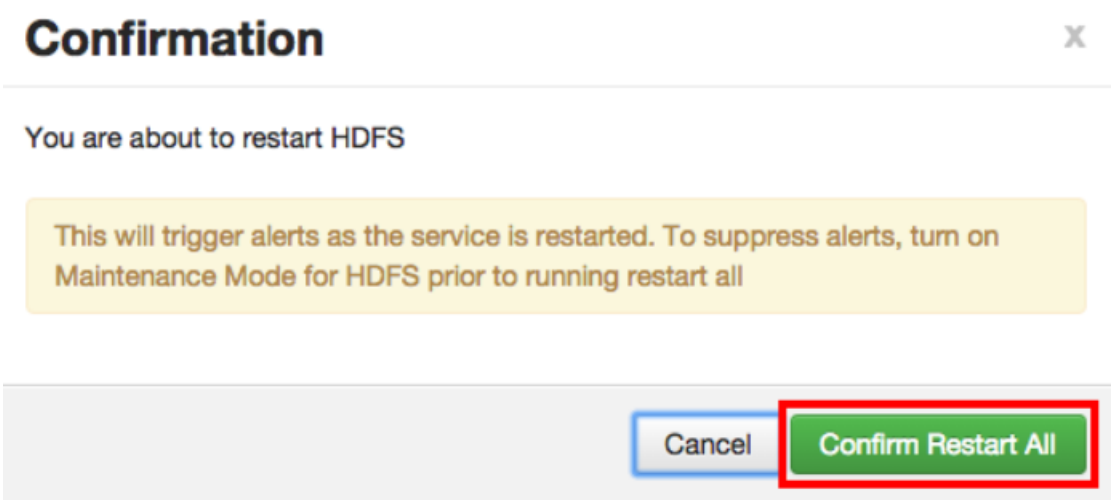


Рис.2.33.: Confirm Restart All

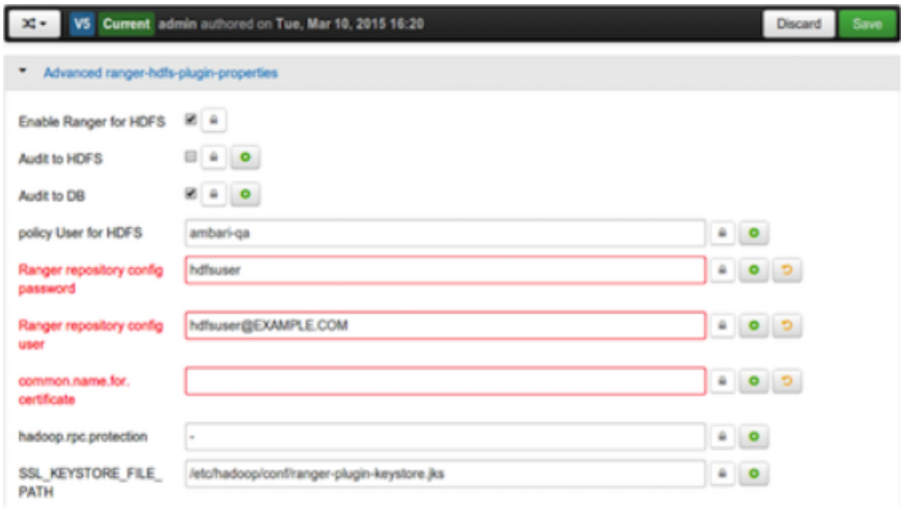


Рис.2.34.: Advanced ranger-hdfs-plugin-properties

Таблица2.11.: Свойства HDFS Plugin

Свойство конфигурации	Значение
Ranger repository config user	rangerhdfslookup@example.com
Ranger repository config password	rangerhdfslookup
common.name.for.certificate	blank

5. После обновления свойств нажать кнопку *Save* и перезапустить сервис HDFS.

2.3 HDFS Policy

2.3.1 Ranger для авторизации в Hadoop

После проверки подлинности пользователя необходимо определить его права доступа. Права доступа пользователя к ресурсам определяет авторизация. Например, пользователю может быть разрешено создание политики и просмотр отчетов, но не разрешено редактирование пользователей и групп. **Ranger** можно использовать для настройки и управления доступом к сервисам **Hadoop**.

Ranger позволяет создавать сервисы для определенных ресурсов **Hadoop** (**HDFS**, **HBase**, **Hive** и др.) и добавлять права доступа к этим сервисам. Можно также создавать сервисы на основе тегов и добавлять политики доступа к ним. Использование политик на основе тегов позволяет управлять доступом к ресурсам нескольких компонентов **Hadoop** без создания отдельных сервисов и политик в каждом компоненте. Можно также использовать **Ranger TagSync** для синхронизации хранилища тегов **Ranger** с внешним сервисом метаданных, таким как **Apache Atlas**.

2.3.2 Создание HDFS Policy

Благодаря конфигурации **Apache Ranger** позволяет проверять для запроса пользователя как политики **Ranger**, так и разрешения **HDFS**. Когда **NameNode** получает пользовательский запрос, плагин **Ranger** проверяет политики, установленные через **Ranger Service Manager**, и если их нет, проверяет разрешения, установленные в **HDFS**.

Рекомендуется создавать разрешения в **Ranger Service Manager** и иметь ограниченные разрешения на уровне **HDFS**.

Добавление новой политики к существующему сервису **HDFS** осуществляется по следующему алгоритму:

1. На странице “Service Manager” выбрать существующий сервис в разделе HDFS (Рис.2.35.).

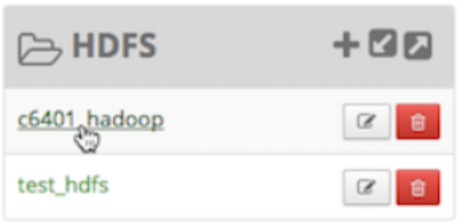


Рис.2.35.: Выбор сервиса HDFS

При этом открывается страница “List of Policies”, на которой необходимо нажать кнопку “Add New Policy” (Рис.2.36.).

2. Открывается страница “Create Policy” (Рис.2.37.).

На странице необходимо заполнить поля. Раздел “Policy Details”:

- *Policy Name* – ввести уникальное имя для данной политики (имя не может быть продублировано нигде в системе);

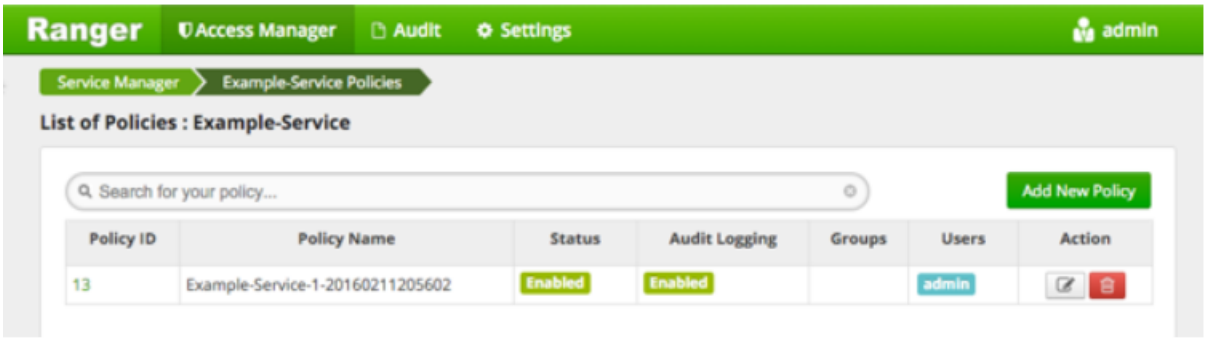


Рис.2.36.: List of Policies

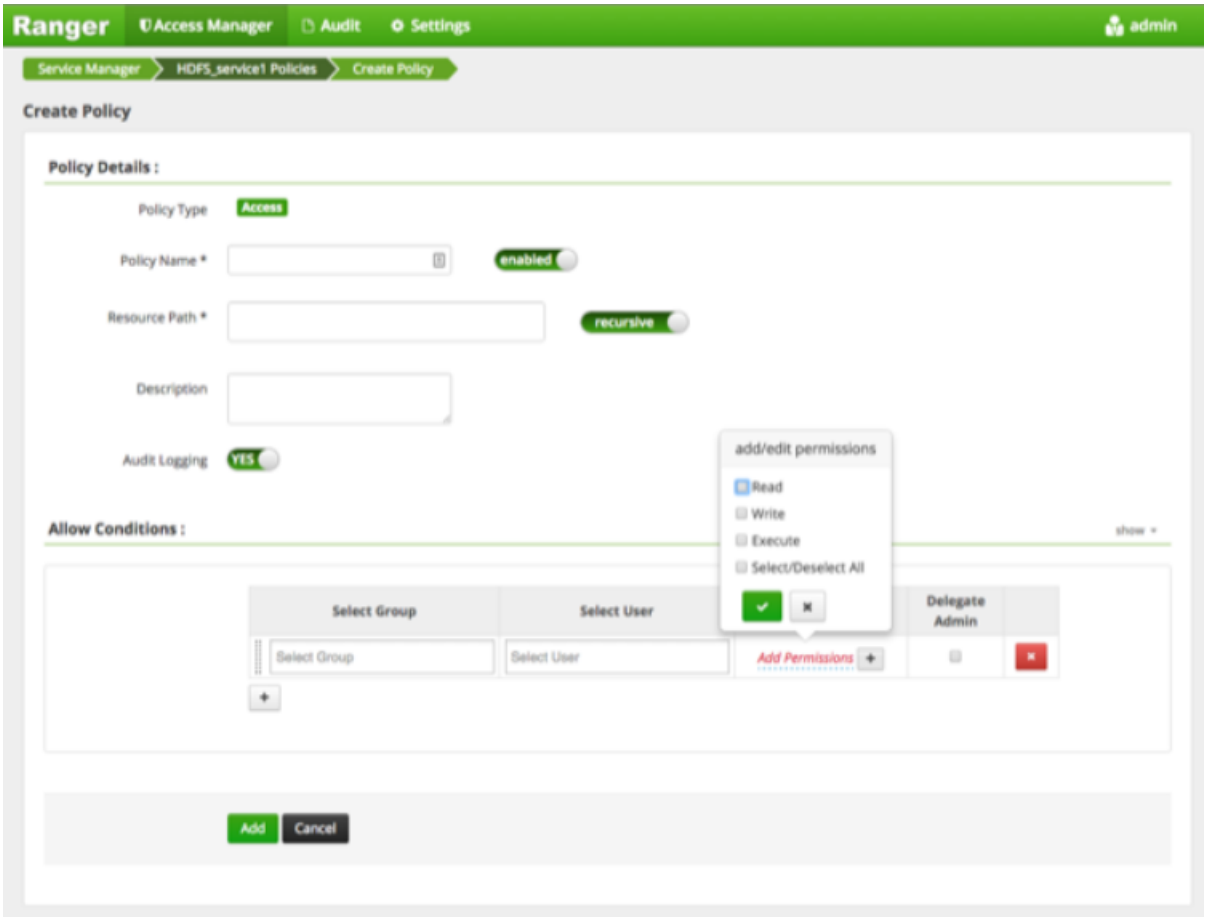


Рис.2.37.: Create Policy

- *Resource Path* – определить путь к ресурсу для папки/файла политики. Во избежание необходимости указывать полный путь или включать политику для всех вложенных папок или файлов, можно заполнить это поле с помощью подстановочных знаков (например, /home*) либо указать, что политика должна быть рекурсивной;
  - Подстановочные знаки могут быть включены в путь ресурса, имя базы данных, таблицы или столбца: "*" – указывает ноль или более символов; "?" – указывает один символ;
- *Description* – (опционально) указать цель политики;
- *Audit Logging* – указать, выполняется ли аудит данной политики (снять флажок, чтобы отключить аудит).

  Раздел "Allow Conditions":

- *Select Group* – указать группу, к которой применяется данная политика. Чтобы назначить группу в качестве администратора для выбранного ресурса, выбрать *Admin permissions* (администраторы могут создавать дочерние политики на основе существующих). Группа *public* содержит всех пользователей, поэтому предоставление доступа к ней предоставляет доступ ко всем пользователям;
- *Select User* – указать конкретного пользователя, к которому применяется данная политика (за пределами уже указанной группы), или назначить определенного пользователя администратором данной политики (администраторы могут создавать дочерние политики на основе существующих);
- *Permissions* – добавить или изменить права: *Read* (чтение), *Write* (запись), *Create* (создание), *Admin* (Администратор), *Select/Deselect All* (выбрать/отменить все);
- *Delegate Admin* – когда политика назначается пользователю или группе пользователей, данные пользователи становятся делегированными администраторами. Делегированный администратор может обновлять, удалять политики. Он также может создавать дочерние политики на основе исходной (базовой);

3. Для добавления дополнительных условий можно использовать символ плюс "+". Условия оцениваются в порядке, указанном в списке – сначала применяется условие в верхней части списка, затем второе, третье и так далее;
4. Нажать кнопку *Add* для сохранения новой политики.