

Информация, необходимая для установки программного обеспечения “Аналитическая платформа Союз”

Содержание

Минимальные требования	1
Инструкция по установке программного обеспечения “Аналитическая платформа Союз”	1
1. Создание серверов.....	2
2. Настройка сервера баз данных	2
3. Настройка сервера приложений	3

Минимальные требования

Минимальные требования к оборудованию, на котором предполагается установка, должно удовлетворять следующим требованиям.

- Количество ядер серверов и оперативной памяти:

Сервер	CPU, ядер	RAM	HDD, Gb
Сервер приложений	4	24	200
Сервер баз данных	8	16	100

Инструкция по установке программного обеспечения “Аналитическая платформа Союз”

1. Создание серверов

1.1. Создать сервер для установки приложений `sppr-app.domain.com` (10.0.1.11) с ОС AstraLinux SE 1.7 Воронеж.

1.2. Создать сервер для установки СУБД `sppr-db.domain.com` (10.0.1.12) с ОС AstraLinux SE 1.7 Воронеж.

1.3. Настроить dns сервер сети ИСППР и файл */etc/resolv.conf* на серверах для корректного разрешения имен серверов по сети.

1.4. Настроить на серверах файл */etc/hosts* для разрешения имен серверов локально.

2. Настройка сервера баз данных

2.1. Установить СУБД PostgreSQL v15 на *sppr-db.domain.com* по инструкции для этого ПО.

2.2. Создать в СУБД пользователя *sppradmin* с правом создания ролей и баз:

```
#su postgres
```

```
#psql
```

```
#CREATE USER sppradmin WITH ENCRYPTED PASSWORD 'password'  
CREATEROLE CREATEDB;
```

2.3. Разрешить пользователю *sppradmin* подключение к *sppr-db.domain.com* от сервера приложений *sppr-app.domain.com* в файле *pg_hba.conf*, указав директиву:

```
host all sppradmin 10.0.1.11/32 md5
```

где *10.0.1.11/32* – ip адрес сервера приложений *sppr-app.domain.com* в сети ИСППР.

2.4. Создать в СУБД пустые базы микро-сервисов, с владельцем *sppradmin*:

audit, auth, analytics, bi, bi_datamart, classification, configuration, core, dictionary, etl_datamart, files, mapcore, notifications, se, sse, settings, workflow;

командой в консоли *psql*:

```
# CREATE DATABASE audit OWNER sppradmin;
```

2.5. Загрузить в базы микро-сервисов схемы из шаблонов или бекапов (выполняется вручную или скриптом инсталляции *set_db.sh* из архива *set_dbisppr.zip*).

2.6. Установить в каталог */opt* на сервере *sppr-db.domain.com*:

- MiniO (S3) version 20230210T184839Z, порты 9000,9001

- MongoDB v6.0.10, порт 27017, используя скрипт *set_mongodb.sh* из архива *set_isppr.zip* и создать в *mongodb* пользователя *zeppelinuser*

- redis-server v7.0.15, порт 6379.

2.7. Создать в ОС sppr-db.domain.com пользователя sppradmin, с правом перезапуска сервисов без пароля, добавив директивы в файл */etc/sudoers*:

```
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl * postgresql-15.service
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl * mongod.service
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl * minio.service
```

2.8. Разрешить в настройках файрвола подключение к серверу sppr-db.domain.com (10.0.1.12) по порту 5432, 6379, 9000,9001, 27017 от сервера sppr-app.domain.com (10.0.1.11).

3. Настройка сервера приложений

3.1. Установить на сервер sppr-app.domain.com:

- OpenJDK Runtime Environment (11.0.17.0.8-3.e17) (build 11.0.17+8-LTS)
- python v3.8.2
- nginx v1.22.1

3.2. Создать в ОС sppr-app.domain.com пользователя sppradmin, с правом перезапуска сервисов ИСППР без пароля, добавив директивы в файл */etc/sudoers*:

```
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl * nifi.service
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl *
zeppelin.service
sppradmin ALL=(ALL) NOPASSWD: /usr/bin/systemctl * sppr-*.service
```

настроить переменные окружения пользователя sppradmin, запустив скрипт *set_sppradmin_env.sh* из архива *set_isppr.zip* или выполнить от имени sppradmin команды:

```
export SYSTEM_USER=gAAAAABlyepeQ
export SIGNING_KEY=4qDvzizr5tMIBIEEQ
export CRYPTO_KEY=o2hzosxraDk9wCc
```

3.3. Установить пользователя `sppradmin` владельцем(рекурсивно) каталогов `/opt`, `/var/log/isppr`.

3.4. Установить в каталог `/opt` на сервере `sppr-app.domain.com`:

- Apache Zeppelin v0.11.1, порт 8080 и настроить переменные окружения в `/opt/zeppelin/conf/zeppelin-env.sh` :

```
export AUTH_URL=http://sppr-app.domain.com:8084
export FILE_SERVICE_URL=http://sppr-app.domain.com:8087
export ZEPPELIN_INTEGRATION_URL=http://sppr-
app.domain.com:8182
export DATAMART_URL=http://sppr-app.domain.com:8185
export ANALYTICS_URL=http://sppr-app.domain.com:8086
export SCHEMA_EDITOR_URL=http://sppr-app.domain.com:8095
export MINIO_URL=http://sppr-app.domain.com:9001
export AUTH_USERNAME=admin
export AUTH_PASSWORD=admin
export ZEPPELIN_ADDR=0.0.0.0
export ZEPPELIN_INTERPRETER_OUTPUT_LIMIT=1024000
export ZEPPELIN_JAVA_OPTS="-Dspark.executor.memory=4g -
Dspark.cores.max=4"
export ZEPPELIN_MEM="-Xms1024m -Xmx4096m -
XX:MaxMetaspaceSize=4096m"
export ZEPPELIN_INTP_MEM="-Xms1024m -Xmx4096m -
XX:MaxMetaspaceSize=4096m"
export ZEPPELIN_LOG_DIR=/var/log/isppr/app/zeppelin
export
ZEPPELIN_NOTEBOOK_STORAGE=org.apache.zeppelin.notebook.re
po.MongoNotebookRepo
export
ZEPPELIN_NOTEBOOK_MONGO_URI=mongodb://zeppelinuser:pass
```

```
word@10.0.1.12:27017/zeppelin.notebook?authMechanism=SCRAM-
SHA-1

export ZEPPELIN_NOTEBOOK_MONGO_DATABASE="zeppelin"
export ZEPPELIN_NOTEBOOK_MONGO_COLLECTION="notebook"
```

- Apache NiFi v1.23.2, порт 443 и загрузить в каталог `/opt/nifi` компоненты `fsd`, используя скрипт `set_fsd.sh` из архива `set_isppr.zip`.

3.5. Настроить запуск от имени `sprradmin` сервисов Zeppelin и NiFi путем создания юнитов в каталоге `/etc/systemd/system`:

`/etc/systemd/system/nifi.service`

```
[Unit]
Description=Apache NiFi daemon
After=syslog.target network.target

[Service]
User=sprradmin
Type=forking
ExecStart=/opt/nifi/nifi-1.23.0/bin/nifi.sh start
ExecStop=/opt/nifi/nifi-1.23.0/bin/nifi.sh stop
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

`/etc/systemd/system/zeppelin.service`

```
[Unit]
Description=Apache Zeppelin daemon
After=syslog.target network.target

[Service]
User=sprradmin
Type=forking
```

```
ExecStart=/opt/zeppelin/bin/zeppelin-daemon.sh start
ExecStop=/opt/zeppelin/bin/zeppelin-daemon.sh stop
ExecReload=/opt/zeppelin/bin/zeppelin-daemon.sh reload
RemainAfterExit=yes
Restart=always
RestartSec=10s
[Install]
WantedBy=multi-user.target
```

3.6. Установить в каталог */opt/front* файлы веб приложения, используя скрипт *set_front.sh* из архива *set_isppr.zip*.

3.7 Установить в каталог */opt/backend* микро-сервисы, используя скрипт *set_back.sh* из архива *set_isppr.zip*. Редактировать переменные окружения каждого микро-сервиса в файле *.env* в соответствии с параметрами серверов, сети, именами баз данных, секретными ключами и учетными записями.

3.8. Настроить запуск от имени *sppradmin* микро-сервисов путем создания юнитов в каталоге */etc/systemd/system*:

- *./sppr-auth.service*
- *./sppr-analytics.service*
- *./sppr-bi-process-task.service*
- *./sppr-mapcore.service*
- *./sppr-notifications.service*
- *./sppr-workflow.service*
- *./sppr-classification.service*
- *./sppr-settings.service*
- *./sppr-se.service*
- *./sppr-files.service*
- *./sppr-bi.service*
- *./sppr-audit.service*
- *./sppr-dictionary.service*

и соответствующих им *socket* файлов

- *./sppr-bi.socket*
- *./sppr-se.socket*
- *./sppr-auth.socket*

- ./sppr-mapcore.socket
- ./sppr-dictionary.socket
- ./sppr-analytics.socket
- ./sppr-classification.socket
- ./sppr-settings.socket
- ./sppr-schema-editor.socket
- ./sppr-audit.socket
- ./sppr-files.socket
- ./sppr-notifications.socket
- ./sppr-workflow.socket

Пример юнита микро-сервиса audit:

```

/etc/systemd/system/sppr-audit.service
[Unit]
Description=sppr audit service
Requires=sppr-audit.socket
After=network.target
[Service]
Type=simple
User=sppradmin
Environment=PROJECT_DIR=/opt/backend/audit
Environment=VENV_DIR=/opt/backend/audit/.venv/bin
WorkingDirectory=/opt/backend/audit
Environment="VIRTUAL_ENV=/opt/backend/audit/.venv"
Environment="PATH=/opt/backend/audit/.venv/bin:/usr/lib/jvm/java-11-
openjdk-11.0.17.0.8-
3.el7.x86_64/bin:/usr/share/Modules/bin:/usr/local/bin:/usr/bin:/usr/local/sbin
:/usr/sbin"
ExecStart=/bin/sh -lc '/opt/backend/audit/.venv/bin/gunicorn --workers=4 --
bind=0.0.0.0:8099 audit.wsgi:application --preload --log-
file=/opt/backend/audit/logs/audit.log --limit-request-field_size 0'
ExecReload=/bin/kill -s HUP $MAINPID
KillMode=mixed
TimeoutStopSec=15
PrivateTmp=true
[Install]
WantedBy=multi-user.target

```

Пример юнита микро-сервиса se:

```

/etc/systemd/system/sppr-se.service
[Unit]

```

```

Description=Gunicorn Daemon for FastAPI Demo Application
Requires=sppr-se.socket
After=network.target
[Service]
User=sppradmin
WorkingDirectory=/opt/backend/se
ExecStart=/bin/sh -lc '/opt/backend/se/.venv/bin/gunicorn -c
gunicorn_conf.py main:app'
ExecReload=/bin/kill -s HUP $MAINPID
KillMode=mixed
TimeoutStopSec=5
PrivateTmp=true
[Install]
WantedBy=multi-user.target

```

3.9. Настроить конфигурационные файлы nginx в каталоге */etc/nginx/conf.d* на работу веб сервера с поддоменами **.sppr-app.domain.com*, соответствующими микро-сервисам, приложениям *zeppelin* и *nifi* и интеграционным сервисам. Настроить соответствие сокетов микро-сервисов портам веб сервера в файле */etc/nginx/nginx.conf*.

Пример для микро-сервиса *audit*:

```

/etc/nginx/conf.d/audit.conf
#audit service
server {
    listen 80;
    listen 443 ssl;
    server_name audit.sppr-app.comitadt.tech;
    ssl_certificate "/etc/pki/nginx/comitadt_tech.crt";
    ssl_certificate_key "/etc/pki/nginx/private/comitadt_tech.key";
    ssl_session_cache shared:SSL:1m;
    ssl_session_timeout 10m;
    ssl_ciphers HIGH:!aNULL:!MD5;
    ssl_prefer_server_ciphers on;
    add_header 'Access-Control-Allow-Origin' always;
    location / {
        # proxy_set_header Host $http_host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_set_header X-NginX-Proxy true;
        proxy_pass http://unix:/run/sppr-audit.sock;
    }
}

```



```
}  
}
```

3.10. Установить в каталог */opt/backend* интеграционные сервисы(java), используя скрипт *set_integration.sh* из архива *set_isppr.zip*.

3.11. Настроить запуск от имени *sppradmin* интеграционных сервисов путем создания юнитов в каталоге */etc/systemd/system*:

- *./nifiIntegration.service*
- *./zeppelinIntegration.service*
- *./taskscheduler.service*
- *./datamart.service*

Пример для сервиса *datamart*:

```
/etc/systemd/system/datamart.service  
[Unit]  
Description=DataMart service  
After=syslog.target network.target  
[Service]  
SuccessExitStatus=143  
User=sppradmin  
Type=simple  
WorkingDirectory=/opt/backend/datamart  
ExecStart=/usr/lib/jvm/java-11-openjdk-11.0.17.0.8-3.el7.x86_64/bin/java -  
jar data-mart.jar  
ExecStop=/bin/kill -15 $MAINPID  
StandardOutput=append:/var/log/datamart_output.log  
StandardError=append:/var/log/datamart_error.log  
[Install]  
WantedBy=multi-user.target
```

3.12. Разрешить в настройках файрвола подключение к серверу *sppr-app.domain.com* (10.0.1.11) по портам 80, 443 из сети пользователей ИСППР.

3.13. Настроить сохранение журналов сервисов:

- sppr-* в каталог */var/log/isppr/backend*
- веб сервера в каталог */var/log/isppr/front*
- интеграционных сервисов в каталог */var/log/isppr/integration*
- zeppelin в каталог */var/log/isppr/app/zeppelin*
- nifi в каталог */var/log/isppr/app/nifi*