ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ИСПОЛЬЗОВАНИЮ МОДУЛЯ METASFERA XD: METASFERA WELDING

Москва
2023 г.

Оглавление

1. Введение

Настоящее руководство описывает процесс установки и настройки модуля Metasfera xD: Metasfera Welding. Подразумевается, что все необходимые компоненты системы размещены в директории /opt, дампы баз данных в директории /opt/db_dumps.

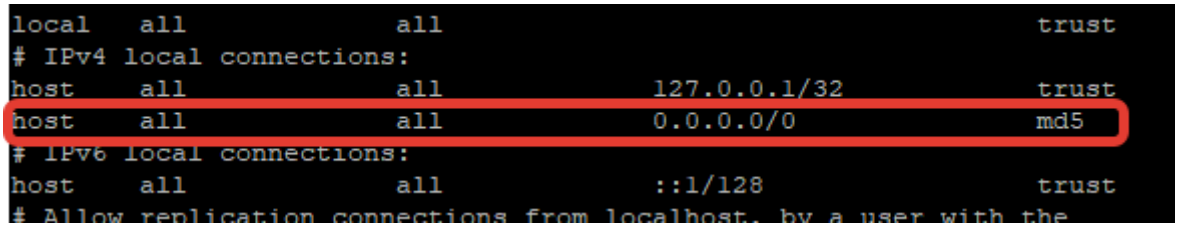
2. Установка и настройка СУБД PostgreSQL

Установка и запуск СУБД PostgreSQL 15:

```
dnf install -y postgresql15-server postgresql15-contrib
postgis-pgsql15 pgagent_15
postgresql-15-setup initdb
systemctl enable postgresql-15 --now
systemctl status postgresql-15
```

Отредактировать файл /var/lib/pgsql/15/data/pg_hba.conf, добавить в ipv4 строку:

```
host  all    all    0.0.0.0/0    md5
```



Перезапустить PostgreSQL:

```
systemctl restart postgresql-15
```

3. Установка и настройка nodejs, npm, pm2

```
dnf install -y nodejs npm
npm install pm2 -g
pm2 startup
```

4. Установка и настройка RabbitMQ

```
dnf -y intall rabbitmq-server
systemctl enable rabbitmq-server --now
systemctl status rabbitmq-server
rabbitmq-plugins enable rabbitmq_management
rabbitmqctl add_vhost consint
rabbitmqctl add_user admin
rabbitmqctl set_user_tags admin administrator
```

```
rabbitmqctl set_permissions -p / admin ".*" ".*" ".*"
rabbitmqctl set_permissions -p consint admin ".*" ".*"
".*"
rabbitmqctl add_user rmq_pmc_user
rabbitmqctl set_permissions -p / rmq_pmc_user ".*" ".*"
".*"
rabbitmqctl set_permissions -p consint rmq_pmc_user ".*"
".*" ".*"
```

5. Установка и настройка Redis

```
dnf install -y redis
```

Настроить /etc/redis/redis.conf:

```
bind 0.0.0.0
protected-mode no
```

Добавить сервис в автозапуск и запустить:

```
systemctl enable redis --now
systemctl status redis
```

6. Установка и настройка nginx

```
dnf install -y nginx
systemctl enable nginx --now
systemctl status nginx
```

7. Установка и настройка tomcat

```
dnf install -y tomcat
systemctl enable tomcat --now
systemctl status tomcat
```

8. Установка и настройка geoserver

```
cd /tmp
wget
https://sourceforge.net/projects/geoserver/files/GeoServe
r/2.20.7/geoserver-2.20.7-war.zip
unzip geoserver-2.23.3-war.zip
mv geoserver.war /var/lib/tomcat/webapps
systemctl restart tomcat
systemctl status tomcat
```

Проверка geoserver, в браузере открыть url: http://server:8080/geoserver

9. Установка и настройка модуля

9.1. Развертывание БД asu_welding

```
su postgres
cd /opt/db_dumps
psql
CREATE USER pmadmin WITH password 'password';
CREATE DATABASE asu_welding WITH OWNER pmadmin;
\c asu_welding
GRANT ALL PRIVILEGES ON SCHEMA log TO pmadmin;
```

9.2. Настройка запуска сервисов с помощью pm2

Создать файл конфигурации */opt/pmc.config.js*:

```
module.exports = {
  apps: [{
        name: "pmc-api",
        script: "node dist/main",
        instances: "1",
        exec_mode: "fork",
        error_file: "log/pmc-api-error.log",
        out_file: "log/pmc-api-out.log",
        cwd: "/opt/pmc-api"
  },
  {
        name: "pmc-security",
        script: "node dist/main",
        instances: "1",
        exec_mode: "fork",
        error_file: "log/pmc-security-error.log",
        out_file: "log/pmc-security-out.log",
        cwd: "/opt/pmc-security"
  },
  {
        name: "pmc-integration",
        script: "node dist/main",
        instances: "1",
        exec_mode: "fork",
```

```
        error_file: "log/pmc-integration-error.log",
        out_file: "log/pmc-integration-out.log",
        cwd: "/opt/pmc-integration"
  }]
}
```

Создать сервисы через pm2:

```
pm2 start /opt/pmc.config.js
```

Записать конфигурацию pm2:

```
pm2 save
```

Проверка сервисов pm2:

```
pm2 list
```

9.2.1. Настройка api

Конфигурация /opt/pmc-api/config/api.yaml:

```
…
host: localhost
httpPort: 4600
…
```

Конфигурация /opt/pmc-api/config/config.yaml:

```
…
import_ksg:
  importDirectory: /opt/uploads/import-ksg
  url: http://'localhost':8099/api/import/mmp
…
smtp:
  email: example@example.ru
  user: example
  password: 123
  host: mail.example.ru
…
reporting:
  reportViewerPath: /opt/uploads/rep
  defaultReportPath: /opt/uploads/rep
  username: user
  password: password
```

```
…
calcschedule_http:
  apiurl: 'http://localhost:8002'
…
pythonRunner:
  path: /opt/pmc-api/python
…
```

Конфигурация /opt/pmc-api/config/db_connection.yaml

```
…
- type: postgres
  database: "asu_welding"
  server: "localhost"
  user: "pmadmin"
  password: "password"
  admin_user: "pmadmin"
  admin_password: "password"
  port: 5432
…
```

Конфигурация /opt/pmc-api/config/db_integration.yaml

```
…
exchange:
  auth:
    user: username
    password: password

  host: https://mail.example.ru
  port: 443
  secure: false
  logger: true
…
```

Конфигурация /opt/pmc-api/config/db_redis.yaml

```
…
  is_use_redis: true
  host: 'localhost'
  port: 6379
```

```
  io_server_port: 4600
…
```

Конфигурация /opt/pmc-api/config/db_upload.yaml

```
---
uploadPath: /opt/uploads
uploadPathComment: Путь, куда сохраняются загруженные
файлы
fileSize: 524288000
fileSizeComment: Максимальный размер загружаемого файла
streamPort: 9999
```

Конфигурация /opt/pmc-api/config/db_microservices.yaml

```
securityService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: security_queue
    prefetchCount: 1
    queueOptions:
      durable: true

integrationService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  queue: integration_queue

kspService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: '5672'
  queue: ksp_queue
```

```
eventService:
  transport: 5
  options:
urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: "event_queue"
    prefetchCount: 100
    noAck: true
    queueOptions:
      durable: true


etl:
  apiBaseUrl: "http://test-dev-all:3000"
  feBaseUrl: "https://app-srv:8091"

loggerService:
  transport: 0
  options:
    host: 'localhost'
    port: 2007

adminService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: security_queue
    prefetchCount: 1
    queueOptions:
      durable: true
```

```
adminNewService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: admin_new_queue
    prefetchCount: 1
    queueOptions:
      durable: true

mainProcessService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  queue: main_process_queue
  prefetchCount: 100
  noAck: true

kspCalcProcessService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  queue: ksp_calc_process_queue
  prefetchCount: 1
  noAck: false

kspProjectProcessService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  queue: ksp_project_process_queue
```

```
  prefetchCount: 1
  noAck: false

kspImportProcessService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
    ]
    queue: "ksp_import_process_queue"
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true

csiProcessService:
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  queue: csi_process_queue
  prefetchCount: 10

coIntegrationService:
  vhost: /
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
    ]
    queue: "co_integration_queue"
    prefetchCount: 100
    noAck: false
```

```
    queueOptions:
      durable: true


documentRenderService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: document_render_queue
    prefetchCount: 1
    queueOptions:
      durable: true
ciService:
[46/1962]
  user: => this.secrets.pmc_rabbit.user
  password: => this.secrets.pmc_rabbit.password
  host: 'localhost'
  port: 5672
  vhost: /
  queue: queue_sync
  prefetchCount: 100
  durable: true


fastReportService:
  host: 'http://localhost'
  port: 10077


intensityScenarioService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
```

```
    queue: intensity_scenario_queue
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true


directiveDateScenarioService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: directive_date_scenario_queue
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true


riskScenarioService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: risk_scenario_queue
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true
socketNewService:
  transport: 5
  options:
    urls: [
```

```
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    queue: socket_new_queue
    prefetchCount: 1
    queueOptions:
      durable: true


qualityScenarioService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
      ]
    prefetchCount: 1
    queue: quality_scenario_queue
    noAck: false
    queueOptions:
      durable: true



documentProcessService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@localhost:5672`'
    ]
    queue: document_process_queue
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true
```

```
reportProcessService:
  transport: 5
  options:
    urls: [
      '=>
`amqp://${this.secrets.pmc_rabbit.user}:${this.secrets.pm
c_rabbit.password}@pmcpg-improve-rabbitmq:5672`
'
    ]
    queue: report_process_queue
    prefetchCount: 1
    noAck: false
    queueOptions:
      durable: true
```

Конфигурация /opt/pmc-api/secret/pmc_pg.yaml

```
---
user: pmadmin
password: password
```

Конфигурация /opt/pmc-api/secret/pmc_rabbit.yaml

```
user: rmq_pmc_user
password: password
```

9.2.2. Настройка security

Конфигурация /opt/pmc-security/config/config.json

```
{
 "kind": "PmcSafetyAppConfig",
 "version": "0.2",
 "title": "конфигурация модуля PmcSafetyAppConfig",
 "installation": "pmcpg-improve",
 "installationType": "develop",
 "flags": [
  "internal_cs"
 ],
 "secret": "Secret",
 "token_expires": "7d",
 "token_expiresComment": "Время жизни токена",
 "time_to_check_expire": 1440,
```

```
 "time_to_check_expireComment": "Время в секундах жизни
токена до его проверки ",
 "main_type_db_connection": "postgres",
 "is_convert_xml": false,
 "api": {
  "title": "Security Microservice",
  "description": "Security Microservices Of API",
  "tag": "PMC-SM",
  "version": "4.0.0",
  "host": "localhost",
  "port": 3000
 },
 "auth": {
  "isLdap": false,
  "isLdapComment": "Проверяет является ли потенциально
авторизованный пользователь, участником LDAP.",
  "isSSO": false,
  "isSSOComment": "Включить/Отключить возможность
авторизации через SSO",
  "isLimitingLoginAttempts": false,
  "isLimitingLoginAttemptsComment": "Включить/Отключить
возможность ограничивать пользователя на попытка ввода
пароля в систему.",
  "isDecryptPassword": false,
  "isDecryptPasswordComment": "Включить/Отключить
шифрования пароля",
  "ldap": [
   {
    "url": "ldap://server:389",
    "baseDN": "DC=domain,DC=local",
    "query": "CN=*",
    "username": "sysdomadm@domain.local",
    "password": "Secret",
    "postfix": "@domain.local"
   }
  ]
 },
 "rabbitmq": {
```

```
  "user": "rmq_pmc_user",
  "password": "password",
  "host": "localhost",
  "port": "5672",
  "queue": "security_queue",
  "prefetchCount": 1,
  "options": {
   "durable": true
  }
 },
 "db_connection": [
  {
   "type": "postgres",
   "database": "asu_welding",
   "server": "localhost",
   "user": "pmadmin",
   "password": "password",
   "port": 5432,
   "connectionTimeout": 60000,
   "schema": "adm",
   "options": {
    "appName": "Metasfera Welding",
    "dateFormat": "dmy",
    "datefirst": 1,
    "language": "русский",
    "encrypt": false
   }
  }
 ]
}
```

9.2.3. Настройка integration

Конфигурация /opt/pmc-integration/config/config.yaml

```
---
kind: PmcIntegrationAppConfig
version: '0.2'
title: Конфигурация для PmcIntegrationAppConfig
```

```
installation: pmcpg-improve
installationType: develop
flags: [ "internal_cs" ]

secret: Sercret

import:
  ksg:
    importDirectory: /opt/uploads/import-ksg
    url: http://localhost:8099/api/import/mmp
    chunkSizeMb: 64
    chunkSizeMbComment: Размер порции файла в МБ, которая используется для импорта графика 2-го уровня
export:
  ksg:
    exportDirectory: /opt/uploads/export-ksg
    url: '?'

documentService:
  render:
    fileDirectory: /opt/uploads/document-service
    url: '?'
```

Конфигурация /opt/pmc-integration/config/db.yaml

```
---
type: postgres
database: 'asu_welding'
schema: 'ksp'
host: 'localhost'
username: 'pmadmin'
password: 'password'
port: 5432
connectionTimeout: 60000
logNotifications: true
applicationName: KSP service
```

Конфигурация /opt/pmc-integration/config/rabbitmq.yaml

```
---
```

```
user: rmq_pmc_user
password: password
host: 'localhost'
port: 5672
…
```

Конфигурация /opt/pmc-integration/config/redis.yaml

```
---
host: 'localhost'
port: 6379
io_server_port: 4600
ttl: 800
ksp_task_ttl: 28800
ksp_task_ttl_description: Время жизни кэш для экрана работ.
ksp_isr_ttl: 600
ksp_isr_ttl_description: Время жизни кэш для экрана ИСР.
ksp_project_ttl: 28800
ksp_project_ttl_description: Время жизни кэш для экрана проектов.
ksp_rsrc_spread_ttl: 28800
ksp_rsrc_spread_ttl_description: Время жизни кэш для экрана распределения ресурсов.
cache_reports_ttl: 1200
cache_reports_ttl_description: Время жизни кэш для отчётов.
```

Конфигурация /opt/pmc-integration/config/dependencies.yaml

```
---
calcschedule-http:
  type: calcschedule-http
  apiurl: 'http://localhost:8002'


modeling-http:
  type: modeling-http
  apiurl: 'http://localhost:8000'

monolith:
```

```
  type: monolith
  apiurl: 'http://localhost:4600'
```

9.3. Установка и настройка модуля Metasfera xD: Metasfera Welding

9.3.1. Развертывание баз данных

```
psql
CREATE USER osm WITH password 'password';
CREATE DATABASE db_gis WITH OWNER osm;
CREATE DATABASE db_model_3d WITH OWNER osm;
CREATE DATABASE db_bim WITH OWNER osm;
CREATE DATABASE db_weld WITH OWNER osm;
exit
pg_restore -d db_gis /opt/db_dumps/db_gis.dump -v
pg_restore -d db_model_3d /opt/db_dumps/db_model_3d.dump
-v
pg_restore -d db_bim /opt/db_dumps/db_bim.dump -v
pg_restore -d db_weld /opt/db_dumps/db_weld.dump -v
```

9.3.2. Установка и настройка модуля отображения 3D-моделей (Geometry)

```
cd opt/geometry/
pm2 start bundle_ws_server.js --name=ws-server --
namespace=ws-server -- --max_old_space_size=32768 --
listen-port=5050
pm2 save
```

9.3.3. Установка и настройка модуля Image server

Создать директорию для хранения изображений:

```
mkdir /srv/storage
```

Создать файл настроек */opt/image_server/app_settings.json*:

```
cat <<\EOF >/opt/image_server/app_settings.json
{
  "auth_settings": {
    "host": "localhost",
    "port": 3000
  },
  "storage_settings": {
    "base_path": "/srv/storage/",
```

```
    "sessions_path": "sessions/"
  },
  "gis_server_settings": {
    "host": "localhost",
    "port": 4502
  },
  "log_settings": {
    "file_name": "/opt/log/image_server.log",
    "product": "image_server"
  },
  "uvicorn_settings": {
    "host": "0.0.0.0",
    "port": 4503
  }
}
EOF
```

Создать сервис системного демона image_server:

```
cat <<\EOF >/etc/systemd/system/image_server.service
[Unit]
Description=Image server service
After=network.target

[Service]
User=root
WorkingDirectory=/opt/image_server
LimitNOFILE=4096
ExecStart=/opt/image_server/image_server
Restart=on-failure
RestartSec=5s

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

Включить и запустить системный демон image_server:

```
systemctl daemon-reload
systemctl enable image_server --now
```

```
systemctl status image_server
```

9.3.4. Установка и настройка модуля Welding GIS

Создать файл настроек */opt/gis/app_settings.json*:

```
cat <<\EOF >/opt/gis/app_settings.json
{
  "db_settings": {
    "host": "localhost",
    "port": 5432,
    "user": "osm",
    "pwd": "password",
    "db_name_gis": "db_gis",
    "db_name_3d": "db_model_3d"
  },
  "auth_settings": {
    "host": "localhost",
    "port": 3000
  },
  "pmc_settings": {
    "host": "localhost",
    "port": 4600
  },
  "geoserver_settings": {
    "host": "localhost",
    "port": 8080,
    "user": "admin",
    "pwd": "password"
  },
  "log_settings": {
    "file_name": "/opt/log/gis.log",
    "product": "gis_core"
  },
  "image_server_settings": {
    "host": "localhost",
    "port": 4503
  },
  "uvicorn_settings": {
```

```
    "host": "0.0.0.0",
    "port": 4502
  }
}
EOF
```

Создать сервис системного демона gis:

```
cat <<\EOF >/etc/systemd/system/gis.service
[Unit]
Description=GIS service
After=network.target

[Service]
User=root
WorkingDirectory=/opt/gis
LimitNOFILE=4096
ExecStart=/opt/gis/gis
Restart=on-failure
RestartSec=5s

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

Включить и запустить системный демон gis:

```
systemctl daemon-reload
systemctl enable gis --now
systemctl status gis
```

9.3.5. Установка и настройка модуля Welding BIM

Создать файл настроек */opt/bim/app_settings.json*:

```
cat <<\EOF >/opt/bim/app_settings.json
{
  "db_settings": {
    "host": "localhost",
    "port": 5432,
    "user": "osm",
    "pwd": "password",
```

```
    "db_name": "db_bim"
  },
  "auth_settings": {
    "host": "localhost",
    "port": 3000
  },
  "log_settings": {
    "file_name": "/opt/log/bim.log",
    "product": "bim_back"
  },
  "uvicorn_settings": {
    "host": "0.0.0.0",
    "port": 4504
  }
}
EOF
```

Создать сервис системного демона bim:

```
cat <<\EOF >/etc/systemd/system/bim.service
[Unit]
Description=BIM service
After=network.target

[Service]
User=root
WorkingDirectory=/opt/bim
LimitNOFILE=4096
ExecStart=/opt/bim/bim
Restart=on-failure
RestartSec=5s

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

Включить и запустить системный демон bim:

```
systemctl daemon-reload
systemctl enable bim --now
```

```
systemctl status bim
```

9.3.6. Установка и настройка модуля Metasfera xD: Metasfera Welding

Создать файл настроек */opt/welding/app_settings.json*:

```
cat <<\EOF >/opt/welding/app_settings.json
{
  "db_settings": {
    "host": "localhost",
    "port": 5432,
    "user": "osm",
    "pwd": "password",
    "db_name": "db_weld"
  },
  "auth_settings": {
    "host": "localhost",
    "port": 3000
  },
  "log_settings": {
    "file_name": "/opt/log/welding.log",
    "product": "welding_back"
  },
  "uvicorn_settings": {
    "host": "0.0.0.0",
    "port": 4505
  }
}
EOF
```

Создать сервис системного демона welding:

```
cat <<\EOF >/etc/systemd/system/welding.service
[Unit]
Description=Welding service
After=network.target

[Service]
User=root
WorkingDirectory=/opt/welding
```

```
LimitNOFILE=4096
ExecStart=/opt/welding/welding

Restart=on-failure
RestartSec=5s

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

Включить и запустить системный демон welding:

```
systemctl daemon-reload
systemctl enable welding --now
systemctl status welding
```

9.3.7. Конфигурация веб-сервера модуля Metasfera xD: Metasfera Welding

Создать файл /etc/nginx/conf.d/welding.conf:

```
server {
    listen       8082;
    server_name  localhost;
    include mime.types;
    client_max_body_size 0;

    location / {
        root   /opt/sites/welding;
        add_header 'Access-Control-Allow-Origin' '*';
        add_header 'Access-Control-Allow-Methods' 'GET,
POST, OPTIONS';
        try_files $uri $uri/ /index.html =404;
    }
    location /weld/ {
        proxy_pass http://localhost:4505;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }
    location /gis/ {
```

```
        proxy_pass http://localhost:4502;
    }
    location /media/ {
        proxy_pass http://localhost:4503;
    }
    location /3d-backend/ {
        proxy_pass http://localhost:5050;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }
    location /file/ {
        proxy_pass http://localhost:4503;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }
    location /bim/ {
        proxy_pass http://localhost:4504;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }
    location /static_sa/ {
        proxy_pass http://localhost:4503;
    }
    location /static/ {
        proxy_pass http://localhost:4503;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
    }
```

```
        location /assets/models_3d/ {
            root /opt/sites/3d-assets/models_3d/;
        }

        error_page   500 502 503 504  /50x.html;
        location = /50x.html {
            root   html;
        }
    }
```

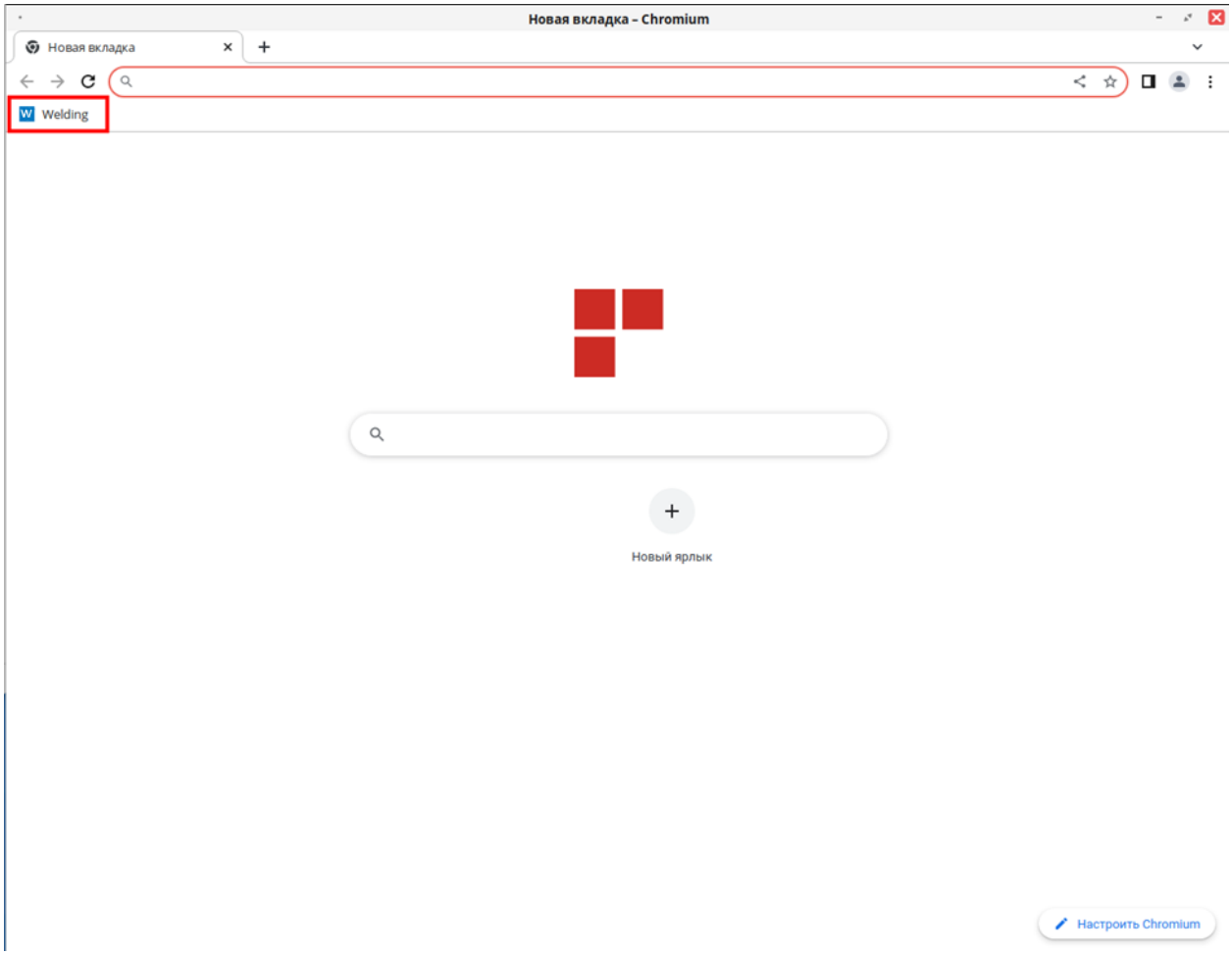
Сделать перезапуск nginx:

```
systemctl restart nginx
```

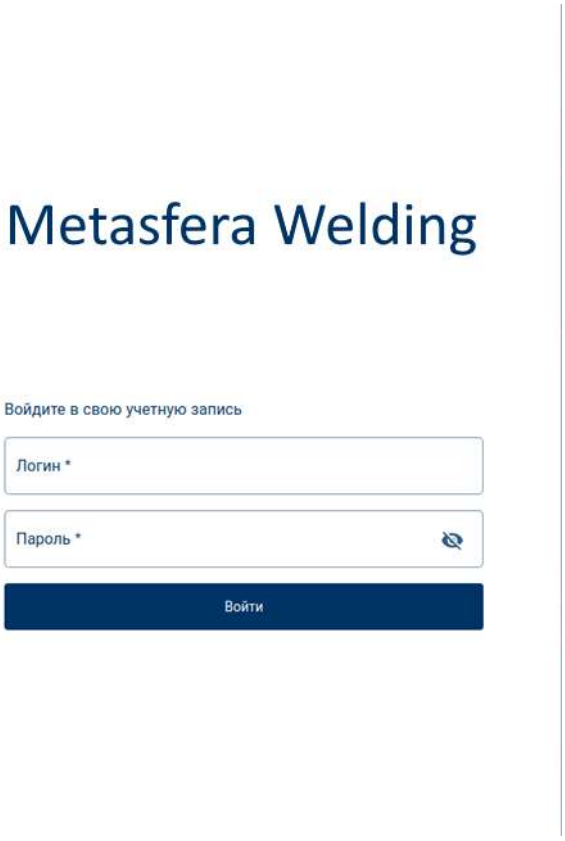
9.4. Запуск браузера

Необходимо нажать на иконку браузера «Chromium» (в данной операционной системе установлен по умолчанию), после чего запустить данный браузер.

По умолчанию открывается окно с логином в систему. Если оно не открылось, выбираем из закладки.



После чего откроется страница логина. На странице логина необходимо ввести логин и пароль от системы.


Metasfera Welding
Войдите в свою учетную запись
Логин *
Пароль *
Войти