



Установка системы

Выпуск 1.0

www.protei-sm.ru

мар. 09, 2023

1	Состав и содержание дистрибутивного носителя данных	1
2	Порядок загрузки данных и программ	2
2.1	Получение дистрибутива	2
2.2	Подготовка к установке	2
2.2.1	Минимальные требования	2
2.2.2	Настройка прокси	3
2.2.3	Предварительные настройки	4
2.3	Установка	5
2.4	Добавление сертификатов	9
2.5	Тестовый запуск интерфейса пользователя	11
2.6	Обновление	12
2.6.1	Обновление установочных файлов	12
2.6.2	Восстановление установочных файлов	14
2.6.3	Обновление сервисов	15
2.6.4	Обновление docker-образов	16
3	Порядок проверки состояния сервисов	17
4	Рекомендации по безопасности	19
4.1	Безопасность операционной системы	19
4.2	Защищённые соединения	19
4.3	Сетевое взаимодействие	19
4.4	Обновление ПО операционной системы	20
4.5	Обновление сервисов PromUC	20

 Состав и содержание дистрибутивного носителя данных

Дистрибутив Promuc поставляется в двух вариантах, различающихся наличием в составе каталога с файлами docker-образов для установки комплекса на оборудовании заказчика: `promuc-debian.tar.gz` (размер менее 1 МБ) и `promuc-full-debian.tar.gz` (размер более 1 ГБ). Других отличий между ними нет.

Таблица 1: Состав дистрибутива

Каталог/файл	Назначение	<code>promuc.tar.gz</code>	<code>promuc-full.tar.gz</code>
<code>install.sh</code>	Скрипт установки	✓	✓
<code>do.sh</code>	Скрипт управления сервисами	✓	✓
<code>update.sh</code>	Скрипт обновления сервиса	✓	✓
<code>upgrade.sh</code>	Скрипт обновления установочных файлов	✓	✓
<code>getimage.sh</code>	Скрипт обновления docker-образа	✓	✓
<code>template.order</code>	Шаблон файла <code>.order</code>	✓	✓
<code>proxy</code>	Каталог сервиса Traefik	✓	✓
<code>postgres</code>	Каталог сервиса Postgres	✓	✓
<code>redis</code>	Каталог сервиса Redis	✓	✓
<code>rabbitmq</code>	Каталог сервиса RabbitMQ	✓	✓
<code>gitea</code>	Каталог сервиса Gitea	✓	✓
<code>framework</code>	Каталог сервиса Module Framework	✓	✓
<code>scada</code>	Каталог сервиса Module SCADA	✓	✓
<code>ruleengine</code>	Каталог сервиса Module RuleEngine	✓	✓
<code>updater</code>	Каталог сервиса Updater	✓	✓
<code>images</code>	Каталог с docker-образами		✓

Порядок загрузки данных и программ

2.1 Получение дистрибутива

Дистрибутив Promuc доступен по следующим ссылкам:

- без docker-образов - <https://download.protei-sm.ru/promuc-debian.tar.gz>
- с docker-образами - <https://download.protei-sm.ru/promuc-full-debian.tar.gz>

Порядок его скачивания будет рассмотрен далее.

2.2 Подготовка к установке

2.2.1 Минимальные требования

- Операционная система: Debian GNU/Linux 11 или иная производная от неё
- Процессор: x64 2 ядра
- Память: 4 ГБ
- Хранилище: не менее 1,2 ГБ
- Дополнительное программное обеспечение:
 - Docker Engine, Compose v2
 - wget, curl, pv, git, bzip2, gnupg, openssl, ca-certificates, lsb-release, httpasswd

2.2.2 Настройка прокси

В случае, если для доступа в Интернет используется прокси, то необходимо предварительно настроить параметры АРТ, docker, профиль пользователя. Пример настройки http-прокси с аутентификацией логин - test.tt, пароль - :T|T^T~T.t2.

- Пример настройки файла /etc/apt/apt.conf.d/99-proxy:

```
Acquire::http::Proxy "http://test.tt::T|T^T~T.t2@10.186.0.116:3128";
Acquire::https::Proxy "http://test.tt::T|T^T~T.t2@10.186.0.116:3128";
```

- Пример настройки файла ~/.profile с добавлением переменных *_proxy:

```
# ~/.profile: executed by Bourne-compatible login shells.

export http_proxy="http://test.tt::T|T^T~T.t2@10.186.0.116:3128"
export https_proxy="http://test.tt::T|T^T~T.t2@10.186.0.116:3128"
export no_proxy="localhost,.local,127.0.0.0/8,promuc.demo,.promuc.demo,172.16.16.0/
↪24"

if [ "$BASH" ]; then
  if [ -f ~/.bashrc ]; then
    . ~/.bashrc
  fi
fi

mesg n 2> /dev/null || true
```

- Пример настройки файла ~/.docker/config.json. В этом файле обязательно заменяем специальные символы их шестнадцатиричным кодом с экранированием символом %:

```
{
  "proxies":
  {
    "default":
    {
      "httpProxy": "http://test%2Ett:%3AT%7CT%5ET%7ET%2Et2@10.186.0.116:3128",
      "httpsProxy": "http://test%2Ett:%3AT%7CT%5ET%7ET%2Et2@10.186.0.116:3128",
      "noProxy": "172.16.16.0/24"
    }
  }
}
```

- Пример настройки файла /etc/systemd/system/docker.service.d/http-proxy.conf. В этом файле специальные символы экранируем двойным символом %:

```
[Service]
Environment="HTTP_PROXY=http://test%2Ett:%3AT%7CT%5ET%7ET%2Et2@10.186.0.
```

(continues on next page)

(продолжение с предыдущей страницы)

```
↪116:3128"
Environment="HTTPS_PROXY=http://test%2Ett:%3AT%7CT%5ET%7ET%2Et2@10.186.0.
↪116:3128"
```

2.2.3 Предварительные настройки

Все описанные далее действия выполняются в командной строке терминала Linux Debian.

В случае отсутствия отдельных компонентов может потребоваться подключение к сети Интернет для их установки.

Установку необходимо производить от имени root или другого пользователя, обладающего аналогичными правами, с командой `sudo`. Чтобы сократить ввод и не добавлять для каждого действия команду `sudo` можно один раз перед установкой выполнить команду `sudo -i` и затем указать пароль учётной записи, чтобы все последующие действия выполнялись с правами суперпользователя или переключиться на пользователя, обладающего правами администратора, например, root, при помощи команды `su - root`:

```
$ sudo -i
[sudo] password for user:
#
```

или

```
$ su - root
Password:
#
```

Каждый сервис Promuc использует отдельный домен следующего уровня. Например, для существующего домена `promuc.loc` сервис Proxu будет использовать адрес `proxu.promuc.loc`.

Для использования сервисов PromUC с других АРМ в сети предприятия настройте в DNS-зоне соответствующие А-записи для сервисов `proxu`, `mq`, `fw`, `git`, `re`

Пример А-записи для Proxu и существующего домена `promuc.loc`:

```
proxu.promuc.loc.      3600    IN      A       185.91.54.107
```

Загрузите и распакуйте соответствующий архив в каталог `/opt`:

```
# cd /opt
# wget https://download.protei-sm.ru/promuc-debian.tar.gz
# tar -xzf promuc-debian.tar.gz
# cd promuc
```

2.3 Установка

Процесс установки будет рассмотрен на примере системы Debian 11, для других версий может потребоваться ручная установка необходимых компонентов, например, Docker.

Установка выполняется скриптом `install.sh`:

```
# ./install.sh --recreate
```

Примечание: при использовании дополнительного ключа `-f` установка будет выполняться без дополнительных запросов после установки каждого сервиса

Внимание: При повторной инсталляции накопленные данные будут уничтожены

Установщик выводит в консоль перечень устанавливаемых сервисов и их очередность установки. Если не был указан ключ `-f`, то для положительного ответа на вопрос можно просто нажать клавишу `Enter`.

```
Выполнить установку PromUC (Y/n): y
Обновляем кеш пакетной базы...

Устанавливаем необходимые дополнительные пакеты curl jq rsync ca-certificates
↳gnupg lsb-release git bzip2 apt-utils apache2-utils

Проверяем наличие и версии Docker и Compose...
Требуется установка Docker
Требуется провести обновление согласно инструкции https://docs.docker.com/engine/
↳install/debian/
Продолжаем? (Y/n): y
```

Далее установщик попытается удалить старые версии имеющихся компонентов:

```
Удаляем docker docker-engine docker.io docker-compose containerd runc
Добавляем репозиторий Docker
Выполняем установку docker-ce docker-ce-cli containerd.io docker-compose-plugin
Согласно файла .order будут установлены следующие компоненты:

proxy
postgres
redis
rabbitmq
framework
gitea
ruleengine
```

(continues on next page)

(продолжение с предыдущей страницы)

```
scada
updater
```

Если необходимо скорректировать то нажмите 'n', отредактируйте данный файл и
↪ повторно запустите установку
Продолжаем? (Y/n):

Далее необходимо ввести адрес домена, в пространстве которого будут функционировать сервисы комплекса (далее для примера будет использоваться домен promuc.loc).

Настраиваем прокси traefik

Введите домен сервисов 2-го уровня или более: promuc.loc

Далее задаем логин администратора (по-умолчанию admin):

Не найден адрес proxy.promuc.loc, добавляем в /etc/hosts

Введите логин администратора: admin

и пароль (генерируется случайный):

Введите пароль администратора (не менее 12 символов): d91cd047c057

Далее скрипт отобразит логин и пароль для панели управления сервиса Traefik и попросит указать адрес электронной почты администратора:

Для доступа к панели мониторинга прокси используйте

URL: https://proxy.promuc.loc

логин: admin

пароль: d91cd047c057

Введите email администратора: admin@promuc.loc

Примечание: Адрес электронной почты, в частности, необходим при использовании PgAdmin4 (средство администрирования базы данных, при необходимости устанавливается отдельно) и должен соответствовать как минимум записи `username@domain.ru` или аналогичной.

Далее выбираем какой сертификат будем использовать для защиты соединений:

Выберите - генерируем сертификат (1), получаем от Let's Encrypt (2) или
↪ использовать собственные (3): 1

1. создание самоверенного сертификата
2. получение публичного сертификата при помощи сервиса в интернете

3. использование имеющихся сертификатов

Для локального развертывания достаточно выбрать первый вариант.

Далее скрипт сообщает расположение файла сертификата ca.crt:

```
Сертификат CA доступен в /opt/promuc/proxy/conf/certs/ca.crt и по ссылке https://
↪promuc.loc/ca.crt
Необходимо импортировать в браузер и в системный список доверенных корневых
↪центров сертификации
```

Указанный сертификат после окончания установки необходимо будет импортировать в список «Доверенных корневых центров сертификации». Например в Яндекс.Браузер - Настройки > Системные > Управление Сертификатами. Аналогичный пункт настройки имеется в других браузерах.

После установки сервиса скрипт выводит результат запуска контейнеров установленного сервиса и запрашивает продолжение установки:

```
Запускаем прокси

Получаем файл ...
.....
.....
[+] Running 2/2
   Network proxy_backend Created
   Container traefik Started
NAME                COMMAND                SERVICE  STATUS  PORTS
traefik             "/entrypoint.sh trae...  traefik  running  0.0.0.0:80->
↪80/tcp, 0.0.0.0:443->443/tcp
Продолжаем? (y/n):
```

Аналогичные сообщения будут выведены после установки остальных сервисов.

По окончании установки создаётся файл /opt/promuc/.config со списком реквизитов доступа к сервисам:

```
Установка Promuc окончена

Основные реквизиты доступа в файле /opt/promuc/.config

### Traefik ###
логин: admin
пароль: 80198eb7
email: admin@promuc.loc

### Сертификат CA ###

-----BEGIN CERTIFICATE-----
```

(continues on next page)

(продолжение с предыдущей страницы)

```
MIIC4jCCAcqgAwIBAgIUFXr9q8xMehI8RDEfGgz8cYc1y2kwDQYJKoZIhvcNAQEL
BQAwFzEVMBMGA1UEAwMZGV2LjRpb3QucHJvMB4XDTIyMDkyMjA5MDQ1NFoXDTIy
MTAyMjA5MDQ1NFowFzEVMBMGA1UEAwMZGV2LjRpb3QucHJvMIIBIjANBgkqhkiG
9wOBAQEFAAOCAQ8AMIIBCgKCAQEAA64Wo0d0DYOpB9OS12IcUgOgVvMfdNa8MnJ+t
vDe90h5sHQd4t4NlyHFhLZP8GMLL86QPI+AUx57gC8jshxZodxhGyTDFJV4wJ8E+
SiWQw3fF17jHeZ0Um+oShxUSUrFw3eNyVgmHrsODvq17A2GhAR6cIZxcIOAAbvq0
5mZ7wkIXCr4wans/bA2B0J91Ve/ADLmqDLhhtdXfcLmcNfZfRFiqq+D+izsfPJBL
/rnw0Ez+yLqY0e87dxDmXiHOaS2gf6uJvFgwIQCsZtNmbzP7IFE2McJs6p01PcxV
yG7FMZDWxrSC+Q4XS147/UvM/lvuW7xbZieQidQacu74H62s0wIDAQABoYJwJDAO
BgNVHQ8BAf8EBAMCAuQwEgYDVR0TAAQH/BAGwBgEB/wIBATANBgkqhkiG9wOBAQsF
AAOCAQEAAJh8ZagfAnIH6gp9XTQXS89Q+itSKzcYFiGuJgH51/zGWQX1dQY+hwKGG
8hRbw5CY6N7kXsQoHoDKBnoLSXfEhEqH9nmDa7FqwUph65YQb1KGC8HcOmd9mbx+
GFNEVxVcJXq6FFMQecbGK/T30FapjbQekt9+dh7pqrUGp1Nh8NUBDQynmQqc7vJb
J2XxNBhrvGGKYjvKi0glzR19DRwFc4NQ44vMB8iAnjIinsHCj5MTdyrCWPkBPt/z
g6E172bpeSnNVoi2ByZHDMbX+Xv0jw==
-----END CERTIFICATE-----
```

```
### Portainer ###
```

```
логин: admin
пароль: 80198eb7
```

```
### PgAdmin ###
```

```
логин: admin@promuc.loc
пароль: 80198eb7
```

```
### RabbitMQ ###
```

```
логин: user
пароль: user
```

```
### Gitea ###
```

```
логин: gitadmin
пароль: 80198eb7
```

```
### RuleEngine ###
```

```
логин: user
пароль: user
```

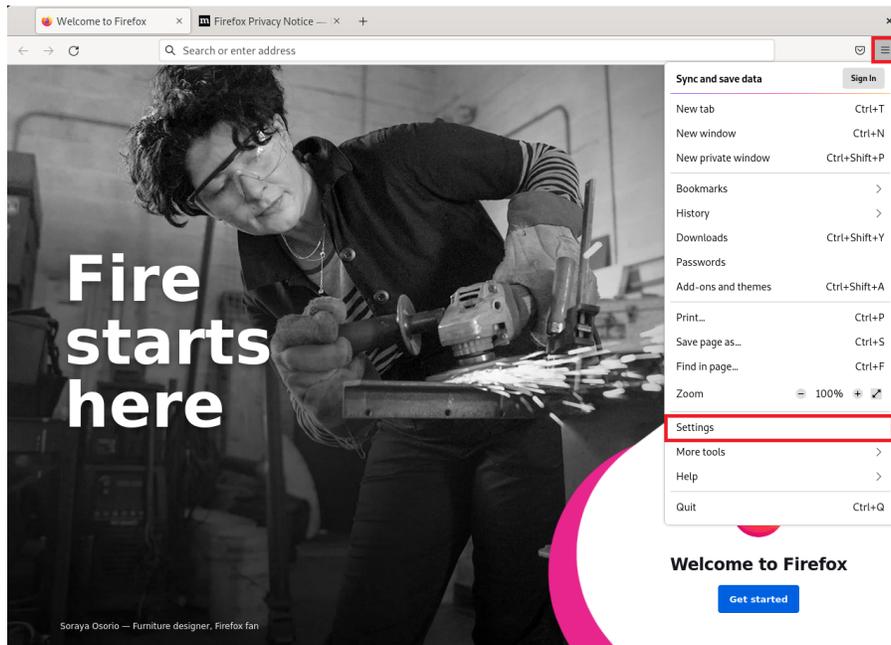
```
### SCADA ###
```

```
логин: gitadmin
пароль: 80198eb7
```

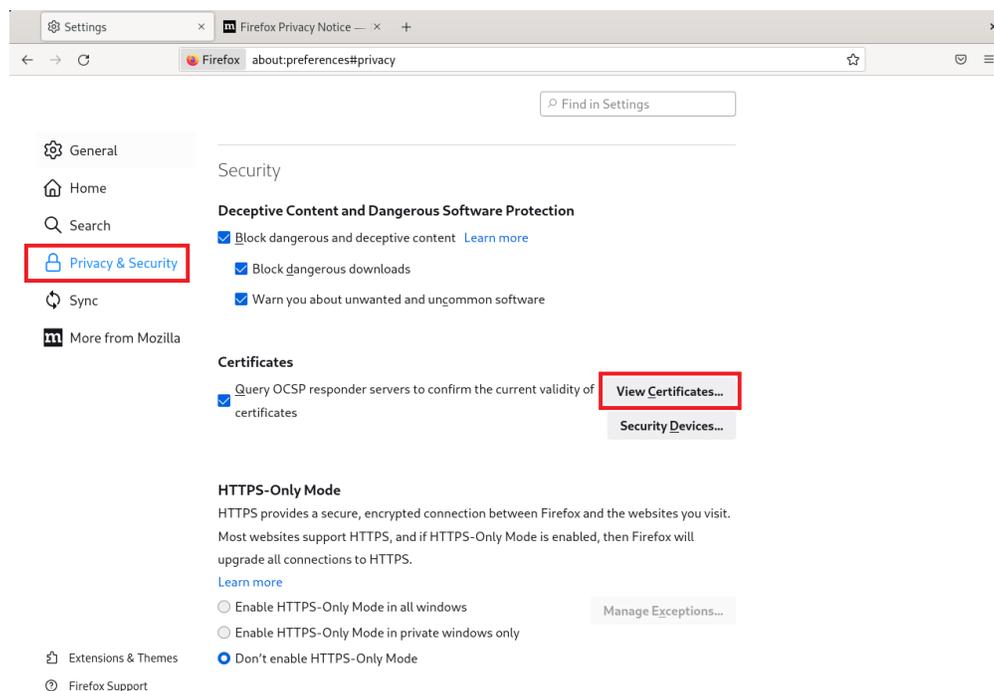
2.4 Добавление сертификатов

Добавление сертификатов будет рассмотрено на примере web браузера Firefox.

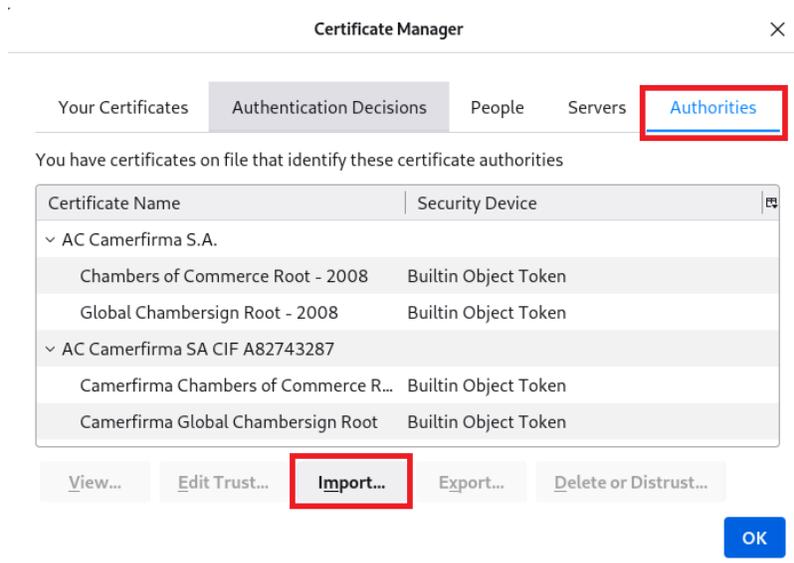
1. Откройте браузер Firefox
2. Кликните на значке меню в правом верхнем углу и затем выберите Settings:



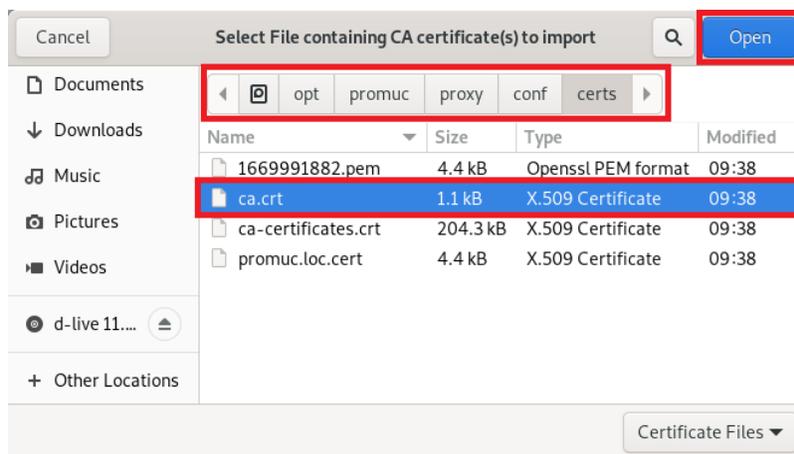
3. Выберите Privacy & Security и затем в разделе Certificates кликните на View Certificates...:



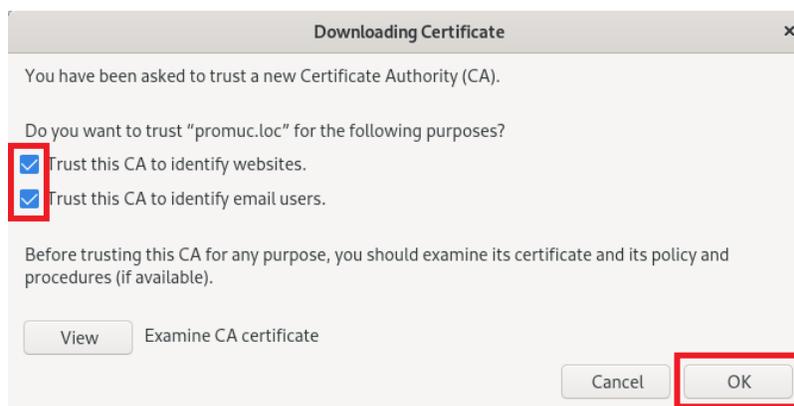
4. Выберите Authorities и кликните на Import...:



5. Перейдите в папку /opt/promuc/proxy/conf/certs и выберите сертификат ca.crt, нажмите Open:

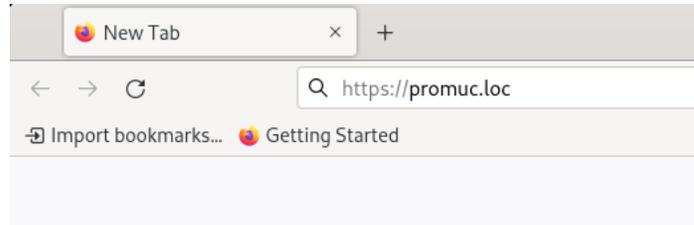


6. Установите флажки и кликните на ОК:

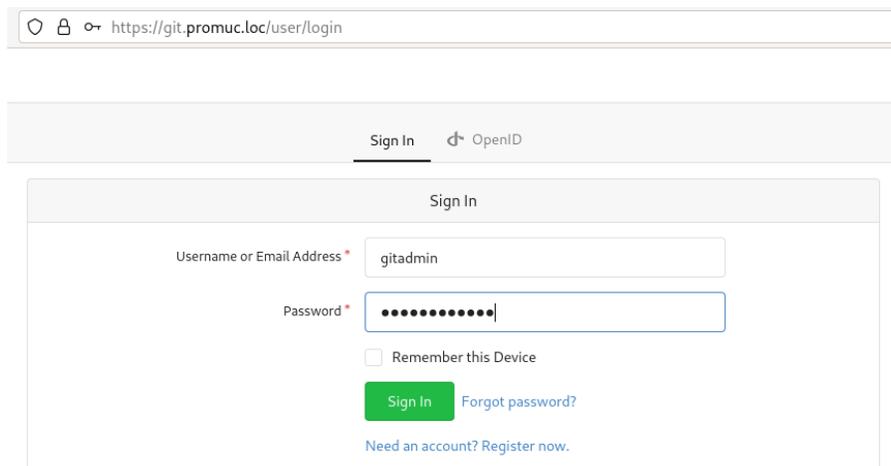


2.5 Тестовый запуск интерфейса пользователя

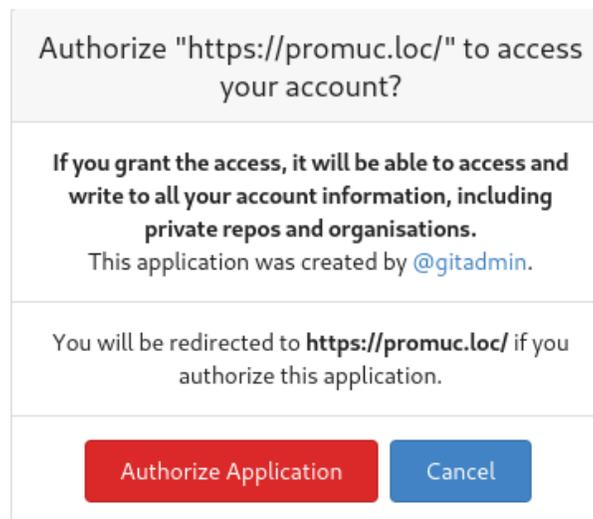
Для запуска интерфейса пользователя необходимо в адресной строке web браузера ввести адрес `https://имядомена`, для данного примера он будет `https://promuc.loc`:



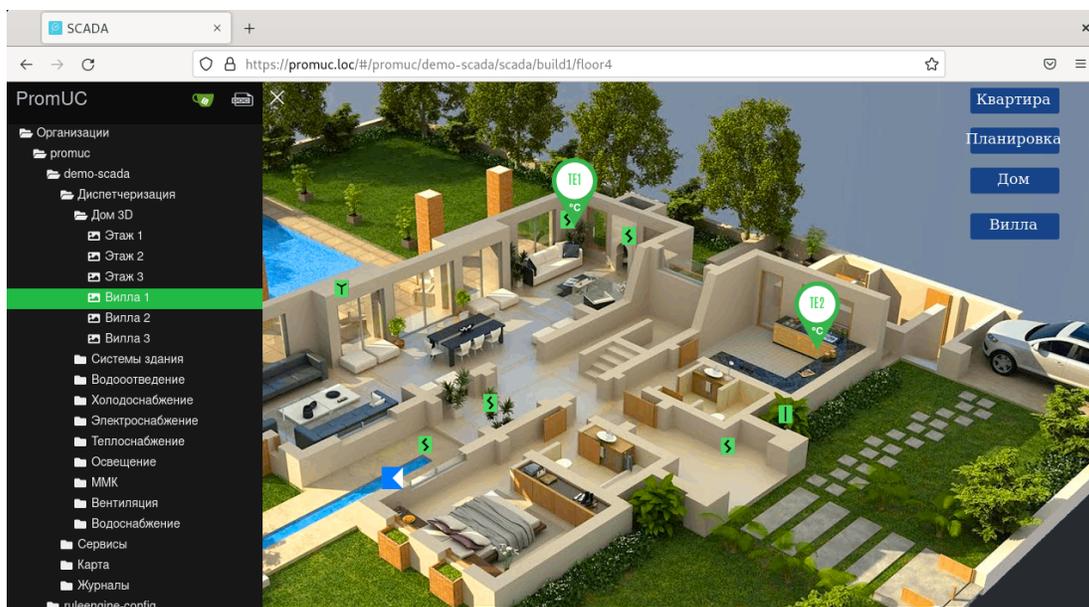
Далее необходимо ввести учётные данные, которые вывел установщик:



и подтвердить авторизацию приложения:



Интерфейс пользователя откроется в окне web браузера:



Сведения о порядке работы с системой можно найти в [Руководстве пользователя](#).

2.6 Обновление

2.6.1 Обновление установочных файлов

Обновление установочных файлов проводится посредством запуска скрипта `upgrade.sh`, который располагается в корневом каталоге дистрибутива, а также его можно скачать с адреса <https://download.promuc.ru/upgrade.sh>:

```
# cd /opt/promuc
# wget --no-check-certificate https://download.promuc.ru/upgrade.{sh,sh.sha256}
# shasum -c upgrade.sh.sha256
upgrade.sh: OK
```

Краткую справку скрипта можно вызвать запустив с ключом `-h`:

```
# /opt/promuc/upgrade.sh -h
Скрипт обновления установки Promuc
Запуск: upgrade.sh -h | -d | [-r] -f <filename>

-f <filename> - файл дистрибутива для обновления
-d - скачать дистрибутив для обновления
-r - восстановление из указанного файла
-h - вызов этой справки
```

Для выполнения обновления скрипту необходимо передать путь и имя файла дистрибутива. Можно использовать как облегчённый вариант дистрибутива, так и полный. Например:

```
# ./upgrade.sh -f ../promuc-debian.tar.gz
или
# ./upgrade.sh -f /opt/promuc-full-debian.tar.gz
```

Если использовать опцию `-d`, то будет выполнено скачивание дистрибутива с сайта:

```
# ./upgrade.sh -d

Внимание! данный скрипт заменит существующие установочные файлы в /opt/promuc и
пересоздаст контейнеры PromUC.
Будет создана резервная копия заменяемых файлов в /opt/promuc/install/.
Для восстановления предыдущей версии используйте upgrade.sh -r -f <имя файла>.

Скачиваем файл дистрибутива, без docker-образов (1) или полный (2): 1
Файл дистрибутива будет сохранён в /opt/promuc-debian.tar.gz
Обнаружен файл дистрибутива /opt/promuc-debian.tar.gz, будет переименован в
/opt/promuc-debian.tar.gz.old
```

В ходе выполнения скрипта проводится сравнение версий установленного комплекса и дистрибутива, сравнение хэш-сумм установочных файлов. Если используется полный дистрибутив, то дополнительно будут сравнены файлы docker-образов сервисов. Перед заменой файлов будет произведено резервное копирование в файл `./install/upgrade_<Дата>.tar.gz`:

```
Получаем дистрибутив с адреса https://download.promuc.ru/promuc-debian.tar.gz
##### 100.0%
Распаковываем дистрибутив в /tmp/upgrade
Исходный файл содержит версию v1.4, используемая версия v1.3.3

Будут заменены файлы:
upgrade.sh
version
doc/install.sh
framework/install.sh
gitea/install.sh
pgadmin/install.sh
portainer/install.sh
proxy/cert.sh
proxy/install.sh
rabbitmq/install.sh
ruleengine/install.sh
scada/docker-compose.yml
scada/install.sh
updater/install.sh

Резервные копии заменяемых файлов будут сохранены в
/opt/promuc/install/upgrade_221101122721.tar.gz
Для их восстановления используйте команду
```

(continues on next page)

(продолжение с предыдущей страницы)

```
/opt/promuc/upgrade.sh -r /opt/promuc/install/upgrade_221101122721.tar.gz
```

Внимание! Требуется пересоздание контейнеров сервисов, т.к. планируются изменения в файлах сервисов или в docker-образах.

Продолжаем? (y/n):

Предупреждение: В случае необходимости обновления установочных файлов сервисов или их файлов docker-образов обязательно будет выполняться пересоздание контейнеров образов, это не затрагивает данные.

При обновлении docker-образов сервисов PromUC будет выполнено сравнение метки даты создания образа и в случае если загружаемый образ создан ранее используемого необходимо будет подтвердить замену. При отказе будет восстановлен ранее используемый образ:

Загружаем обновлённые docker-образы...

Обновляем docker-образ psm/promuc-scada

Уже имеется загруженный docker-образ psm/promuc-scada:latest с id a630502d9a86,

↪ проверяем даты создания...

The image psm/promuc-scada:latest already exists, renaming the old one with ID

↪ sha256:a630502d9a8634abff2c200861cd071a102a1df34c0f5d77e7b576ebd5dd370 to empty

↪ string

Loaded image: psm/promuc-scada:latest

Внимание! В файле содержится образ psm/promuc-scada предыдущей версии.

Использовать его? (y/n): n

Образ a630502d9a86 восстановлен в psm/promuc-scada:latest

Файл docker-образа восстановлен в /opt/promuc/images/psm_promuc-scada.bz2

По окончании будет предложено запустить сервисы если они были остановлены во время обновления.

2.6.2 Восстановление установочных файлов

Для отката изменений, восстановления установочных файлов необходимо запустить скрипт upgrade.sh с ключами -r -f <filename>:

```
# ./upgrade.sh -f /opt/promuc/install/upgrade_221101123321.tar.gz -r
```

Будет выполнено восстановление файлов из резервной копии

/opt/promuc/install/upgrade_221101130531.tar.gz в /opt/promuc.

Контейнеры сервисов будут пересозданы.

Продолжаем? (y/n): y

Проверяем целостность архива...

(continues on next page)

(продолжение с предыдущей страницы)

```

Будут восстановлены файлы:
doc/install.sh
framework/install.sh
scada/install.sh
version
Будут восстановлены docker-образы:
images/psm_ruleengine-transport.bz2
images/psm_promuc-doc.bz2
Продолжаем? (y/n):

```

Примечание: При восстановлении резервное копирование не выполняется.

2.6.3 Обновление сервисов

Обновление сервисов проводится посредством запуска скрипта `update.sh` с указанием конкретного сервиса (папка сервиса). Например:

```

# ./update.sh scada

Получаем образ psm/promuc-scada
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 39.4M  100 39.4M    0     0  23.6M      0  0:00:01  0:00:01  ---:---:-- 23.6M
Loaded image: psm/promuc-scada:latest

Рестартуем сервис scada

Run 'up -d' for scada:
[+] Running 1/0
   Container scada  Running

```

Скрипт получает docker-образ с сайта <https://download.protei-sm.ru> посредством скрипта `getimages.sh`, сохраняет в папку `images` и пересоздаёт контейнер сервиса с новым образом.

Примечание: Если полученный с сайта образ соответствует используемому, то контейнер не пересоздаётся.

2.6.4 Обновление docker-образов

В случае необходимости проведения самостоятельного или поэтапного обновления возможно получение отдельного обновлённого docker-образа сервиса с сайта <https://download.protei-sm.ru>. В этом случае полученный файл docker-образа сохраняется с заменой существующего файла в папке images и подгружается в docker, контейнер сервиса при этой операции не пересоздаётся.

Для получения образа необходимо запустить скрипт getimages.sh с указанием наименования docker-образа, нескольких образов (разделяя пробелом) или без указания для получения образов всех сервисов. Например:

```
root@vpn1:/opt/services# ./getimage.sh -p psm/promuc-framework psm/promuc_scada

Получаем образ psm/promuc-framework
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 39.4M  100 39.4M    0     0  88.9M      0  ---:---:--  ---:---:--  ---:---:--  88.7M
Loaded image: psm/promuc-framework:latest

Получаем образ psm/promuc_scada
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 144M  100 144M    0     0  71.4M      0  0:00:02  0:00:02  ---:---:--  71.4M
Loaded image: psm/promuc_scada:latest
```

Примечание: Флаг `-p` включает принудительную загрузку файла docker-образа с сайта независимо от наличия в папке images.

Таблица 1: Используемые docker-образы

Сервис	Файл docker-образа
proxy	traefik:v2.7
postgres	postgres:14
redis	redis:7.0.0
rabbitmq	rabbitmq:3.10.1-management
framework	psm/promuc_framework
gitea	gitea/gitea:1.17.3
ruleengine	psm/promuc_flow psm/ruleengine psm/ruleengine-transport psm/ruleengine-ws
scada	psm/promuc_scada
updater	psm/updater

Порядок проверки состояния сервисов

Чтобы проверить состояние сервисов можно воспользоваться стандартными командами docker или скриптом do.sh:

```
# ./do.sh scada ps
Run 'ps' for scada:
NAME                COMMAND                SERVICE                STATUS                ↴
↪ PORTS
doc                  "/docker-entripoint...."  scada                  running              ↴
↪ 80/tcp
```

```
# ./do.sh scada logs
Run 'logs' for scada:
scada | /docker-entripoint.sh: /docker-entripoint.d/ is not empty, will attempt
to perform configuration
scada | /docker-entripoint.sh: Looking for shell scripts in /docker-entripoint.d/
scada | /docker-entripoint.sh: Launching /docker-entripoint.d/10-listen-on-ipv6-by-
↪ default.sh
scada | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/
↪ conf.d/default.conf
scada | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/
↪ nginx/conf.d/default.conf
scada | /docker-entripoint.sh: Launching /docker-entripoint.d/20-envsubst-on-
↪ templates.sh
scada | /docker-entripoint.sh: Launching /docker-entripoint.d/30-tune-worker-
↪ processes.sh
scada | /docker-entripoint.sh: Configuration complete; ready for start up
scada | 2022/09/22 09:07:51 [notice] 1#1: using the "epoll" event method
```

(continues on next page)

(продолжение с предыдущей страницы)

```
scada | 2022/09/22 09:07:51 [notice] 1#1: nginx/1.21.6
scada | 2022/09/22 09:07:51 [notice] 1#1: built by gcc 10.3.1 20211027 (Alpine 10.
↪3.1_git20211027)
scada | 2022/09/22 09:07:51 [notice] 1#1: OS: Linux 5.10.0-16-amd64
scada | 2022/09/22 09:07:51 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
scada | 2022/09/22 09:07:51 [notice] 1#1: start worker processes
scada | 2022/09/22 09:07:51 [notice] 1#1: start worker process 32
scada | 2022/09/22 09:07:51 [notice] 1#1: start worker process 33
```

4.1 Безопасность операционной системы

В целях обеспечения безопасности комплекса PromUC рекомендуются ресурсы:

- О РЕКОМЕНДАЦИЯХ ПО ОБЕСПЕЧЕНИЮ БЕЗОПАСНОЙ НАСТРОЙКИ ОПЕРАЦИОННЫХ СИСТЕМ LINUX ФСТЭК России от 30 декабря 2022 г. N 240/22/6933 <https://fstec.ru/normotvorcheskaya/informatsionnye-i-analiticheskie-materialy/2590-informatsionnoe-soobshchenie-fstek-rossii-ot-30-dekabrya-2022-g-n-240-22-6933>

4.2 Защищённые соединения

Для взаимодействия с сервисами PromUC используется протокол HTTPS. В промышленной среде рекомендуется использовать выделенный сервис «Центр сертификации». PromUC позволяет использовать «ключевую пару» сформированную в инфраструктуре заказчика.

4.3 Сетевое взаимодействие

Рекомендуется использование PromUC в выделенном сетевом сегменте. Допускается применение локального программного файрвола с использованием ПО iptables, ufw и др. Используемые порты и протоколы:

- 443/tcp
- 15675/tcp
- 15676/tcp

- 1883/tcp
- 5555/tcp
- 7777/tcp
- 80/tcp (для получения и обновления сертификатов с внешнего сервиса)
- 502/tcp, 47808/udp, 161-162/udp, 3671/udp (включаются в зависимости от конфигурации RuleEngine)

Порты используемые для межконтейнерного взаимодействия, не публикуемые для внешнего доступа:

- 5432/tcp
- 6379/tcp
- 9000/tcp

4.4 Обновление ПО операционной системы

Для эксплуатации в промышленной среде рекомендуется использовать операционную систему с длительным сроком сопровождения, обеспечивающее регулярные обновления безопасности ПО. Регулярно выполняйте обновление программного обеспечения операционной системы с установленным комплексом Promis. Рекомендуется предварительно выполнять резервное копирование данных Promis и останавливать сервисы PromUC.

```
./do.sh -r --all down  
apt update  
apt upgrade  
./do.sh --all up -d
```

4.5 Обновление сервисов PromUC

Рекомендуется регулярно проводить обновление сервисов PromUC.

Используемая версия docker-образов инфраструктурных сервисов, таких как Proxy, Postgres, RabbitMQ, Redis, PGAdmin, Portainer, указывается в файле `.env` каталога сервиса в переменной с префиксом `_TAG`. Например:

```
# cat redis/.env  
REDIS_TAG=7.0
```

Для обновления docker-образа необходимо изменить тэг и выполнить команду:

```
./update.sh redis
```

Примечание: Для успешного обновления необходим доступ к сети Интернет

Для обновления сервисов таких как, FrameWork, Scada, Ruleengine, Updater, использующих docker-образы с префиксом psm/ не требуется указания тега, только выполнение команды `./update.sh <сервис>`.