Инвентаризация

Инвентаризация: Настройка

Авторизация

Приложение поддерживает следующие режимы авторизации пользователей:

- **Отсутствие контроля прав** - в таком режиме можно выполнять все действия может любой человек получивший веб доступ (ограничения могут быть настроены на уровне авторизации веб сервера или ограничения по IP). Приложение работает в таком режиме по умолчанию
- **Требование аутентификации для работы** - для доступа к данным нужно будет пройти аутентификацию, выполнив «Вход». Для этого должна быть настроена аутентификация и выставлен параметр

```
'authorizedView'=>false,      //аутентификация достаточна для авторизации
```

  - false - (по умолчанию) не требовать аутентификацию,
  - true - требовать аутентификацию для работы
- **RBAC** - Role Based Access Control, в таком режиме каждому пользователю необходимо задавать разрешения, на просмотр, редактирование и изменение доступа пользователей. Для этого должна быть настроена аутентификация и выставлен параметр

```
'useRBAC'=>false,             //включить  RBAC  и выдавать права только тем кому они явно назначены
```

  - false - (по умолчанию) ничего не требовать - у всех полные права
  - true - права назначаются явно каждому пользователю, по умолчанию никто ничего не может (если включено authorizedView, то авторизованные получают роль viewer)

Комбинации аутентификации и RBAC:

authorizedView	useRBAC	результат
false	false	аутентификации, авторизации и ограничений в правах нет - всем можно все
true	false	аутентификация требуется, полные права у всех аутентифицированных
false	true	аутентификация не требуется, но без нее права только на просмотр, остальные нужно выдавать явно и они будут предоставлены только после аутентификации
true	true	аутентификация требуется, и никаких прав по умолчанию нет, все права нужно выдавать явно

Аутентификация

Чтобы пользователь мог вводить пароль, надо их где-то хранить, вариантов всего два

Local backend

Хэши паролей зашифрованные Blowfish будут храниться прямо в БД. Для включения такого режима нужно выставить параметр

```
'localAuth'=>true,            //включить локальную БД паролей
```

LDAP backend

Для этого локальное хранение паролей должно быть отключено:

```
'localAuth'=>false,           //включить локальную БД паролей
```

Настройки LDAP указываются следующим образом: нужно создать файл config/ldap.php следующего содержания:

```
<?php
return [
    'class'=>'Edvlerblog\Adldap2\Adldap2Wrapper',
    'providers'=> [
        'default'=>[
            'autoconnect'=>true,
            'config'=>[
                'ad_port'      => 636,
                //'ad_port'      => 389,
                'hosts'    => ['dc1.domain.local','dc2.domain.local'],
                'account_suffix' =>  '@domain.local',
                'base_dn' => "DC=domain,DC=local",
                //под кем подключиться к АД (подойдет любой пользователь. права
admина не нужны)
                'username' => 'inventory@domain.local',
                'password' => 'SuperSecretPassword1!',
                'use_ssl'   => true,
                'use_tls'   => true,
                'custom_options'   => [
                    // See: http://php.net/ldap_set_option
                    LDAP_OPT_X_TLS_REQUIRE_CERT => LDAP_OPT_X_TLS_NEVER
                ],
            ],
        ],
    ],
];
```

Быстрый старт

- Стартуем приложение без контроля доступа

```
'authorizedView'=>false,
'useRBAC'=>false,
```

- Заводим пользователя вручную (на этом этапе для этого не нужно ни полномочий ни авторизации), допустим это пользователь **pupkin**
- Настраиваем аутентификацию (включаем локальную или настраиваем LDAP)
- В случае локальной аутентификации создаем(сбрасываем) пароль созданному пользователю **pupkin**
- Включаем RBAC

```
'useRBAC'=>true,
```

- Выдаем **pupkin** права админа из консоли

```
php yii rbac/grant admin pupkin
```

либо при работе в докере контейнере

```
docker exec -it "armsdocker-app-arms-app-1" php yii rbac/grant admin
pupkin
```

- Авторизуемся под **pupkin**
- ???
- PROFIT

Params

Дополнительные параметры определяются в файле config/params.local.php в формате

```
<?php
return [
    'param'=>'value',
];
```

Авторизация/аутентификация

- **authorizedView**: *true*|*false* требовать авторизации для работы с системой
- **useRBAC**: *true*|*false* использовать разграничение прав на базе ролей
- **localAuth**: *true*|*false* использовать локальную БД паролей для авторизации (иначе LDAP)

Настройки генератора номеров оборудования

- **techs.invNumStrPads**: количество знаков числовой части инвентарного номера в зависимости от количества префиксов. Default:

```
[9,6,4,4]
```

- без префикса 9, с одним префиксом 4, с двумя 4, дальше также 4...

Пример

```
000000001 //0 префиксов - 9 цифр
МСК-000001 //1 префикс - 6 цифр
МСК-ПРН-0001 //2 префикса - 4 цифры
МСК-IT-ПРН-0001 //3 префикса также 4 цифры
```

- **techs.prefixFormat**: формат автоматического формирования префикса инв. номера. Default:

  ```
  ['place','org','type']
  ```

  - сначала префикс помещения, потом организации-владельца оборудования, потом тип оборудования

- **techs.invNumMaxLen**: максимальная длина в которую будет пытаться уложиться формирование инв. номера уменьшая по возможности. Default: **15** (Внимание, макс. размер поля - 16 знаков)

Дополнительная маркировка

- **techs.uidLabel**: название дополнительной маркировки оборудования. Напр. QR-код, Штрих-код. Default: **Доп. маркировка**
- **techs.uidHint**: подсказка для заполнения дополнительной маркировки оборудования. Default: **Какая-либо дополнительная маркировка нанесенная на оборудование**

Сети

- **networkDescribeSegment**: показывать описание сегмента сети на страничке сети. Default: **auto**
  - true - да
  - false - нет
  - auto - если нет описания самой сети
- **networkInlineDescriptionLimit**: максимальное количество строк описания сети которое выводить на прямо на страничку. Все что больше будет скрыто во вкладку на страничке. Default: **20**

Подразделения

- **departments.enable**: включить использование функциональных подразделений (если оргструктура организаций не отображает функционального разделения на отделы)
  - true - включить
  - false - выключить (по умолч.)

Ответственные/поддержка

- **support.service.min.weight**: минимальный вес сервиса, необходимый для учета его команды как поддержка узла/оборудования на котором он живет. Например если задать тут 20, то команда сервиса весом в 10 не будет считаться поддержкой серверов и оборудования сервиса (предполагается что там должны быть более весомые сервисы для этого)

Софт

- **soft.deferred_rescan**: отложенный рескан софта: в случае изменения состава ПО необходимо пересканировать все ОС на предмет наличия там этого ПО. В случае большого количества ОС - процедура весьма медленная.
    - true - отложенный режим: при изменении элемента ПО в справочнике ПО только список ОС для rescan. Необходимо через cron регулярно вызывать команду yii comps/rescan, которая будет сканировать по 100 ОС за раз
    - false - realtime режим: при изменении элемента ПО в справочнике ПО все ОС будут проверены заново на состав ПО.

Интеграция с Wiki

- **wikiUrl**: адрес dokuWiki для интеграции
- **wikiUser**: логин для подключения к dokuWiki
- **wikiPass**: пароль для к dokuWiki

From:
http://wiki.reviakin.net/ - **Wiki**

Permanent link:
http://wiki.reviakin.net/%D0%B8%D0%BD%D0%B2%D0%B5%D0%BD%D1%82%D0%B0%D1%80%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F:%D0%BD%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0?rev=1741762571

Last update: **2025/03/12 06:56**