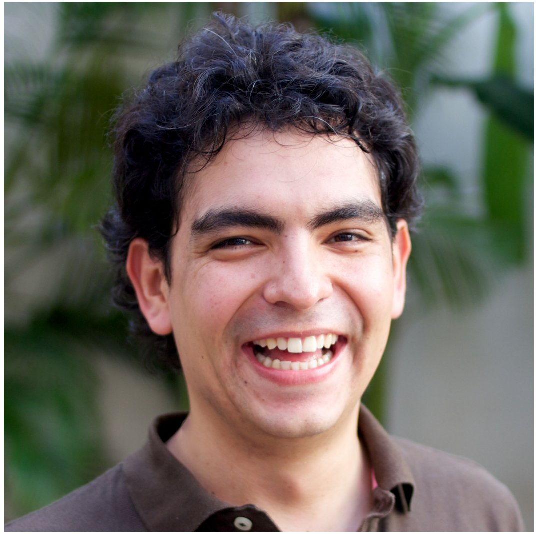
Juampy Novillo Requena

Authentication in Drupal

DrupalCamp Spain 2014

About me, @juampy72

Drupal 7 and 8 module maintainer and core developer



Developer at Lullabot

Let's start by defining Authentication and Authorization

Authentication

Show me your ID, sucker!

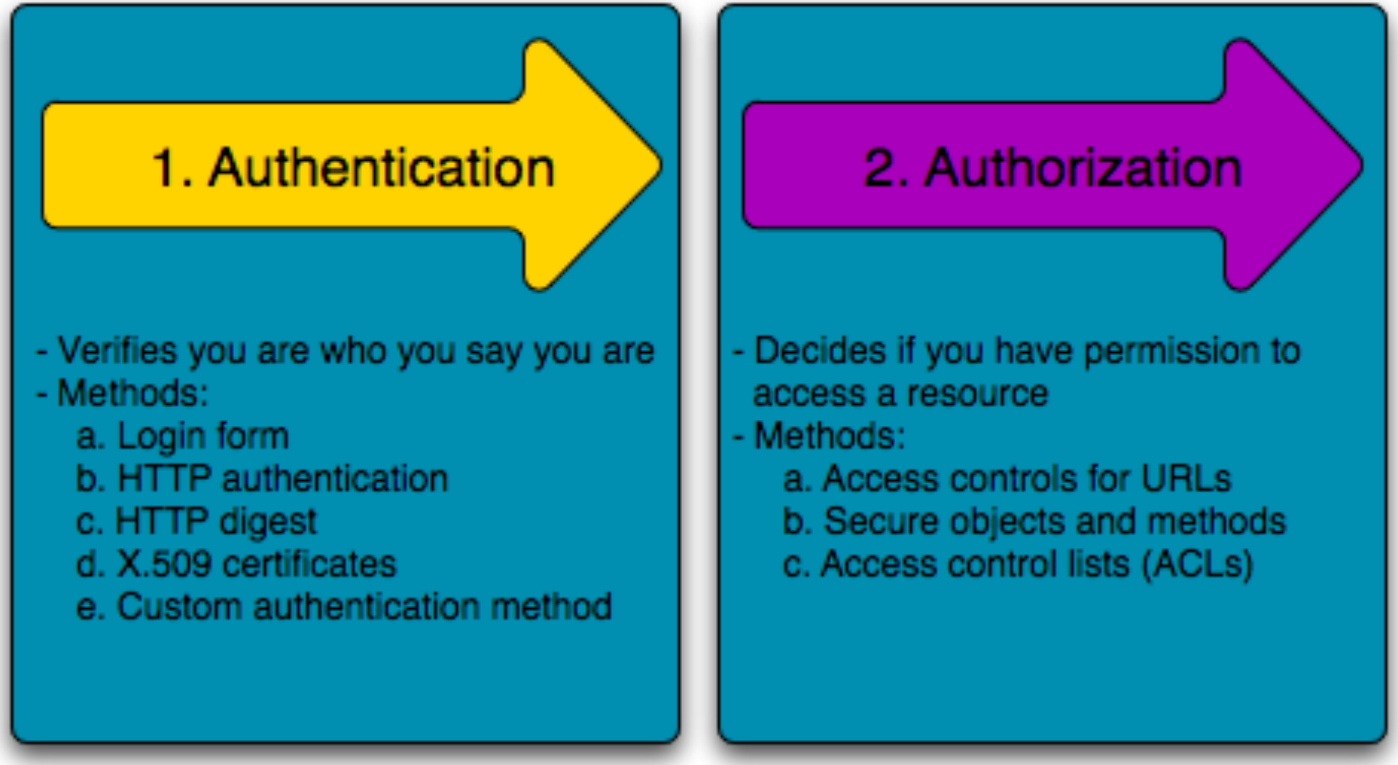
http://aunquedamuchoquedecir.files.wordpress.com/2013/10/entrada-rave-market-madrid-en-sal[illegible]=560

Authorization

403

None shall pass!!

As the Symfony book states...



1. Authentication

- Verifies you are who you say you are
- Methods:
  a. Login form
  b. HTTP authentication
  c. HTTP digest
  d. X.509 certificates
  e. Custom authentication method

2. Authorization

- Decides if you have permission to access a resource
- Methods:
  a. Access controls for URLs
  b. Secure objects and methods
  c. Access control lists (ACLs)

http://symfony.com/doc/current/book/security.html

Authentication in Drupal 8

Drupal 8 implements a Modular Authentication System.

Different Authentication Providers may extract a Drupal **$user** out of a given **$request**.

Auth Providers in core

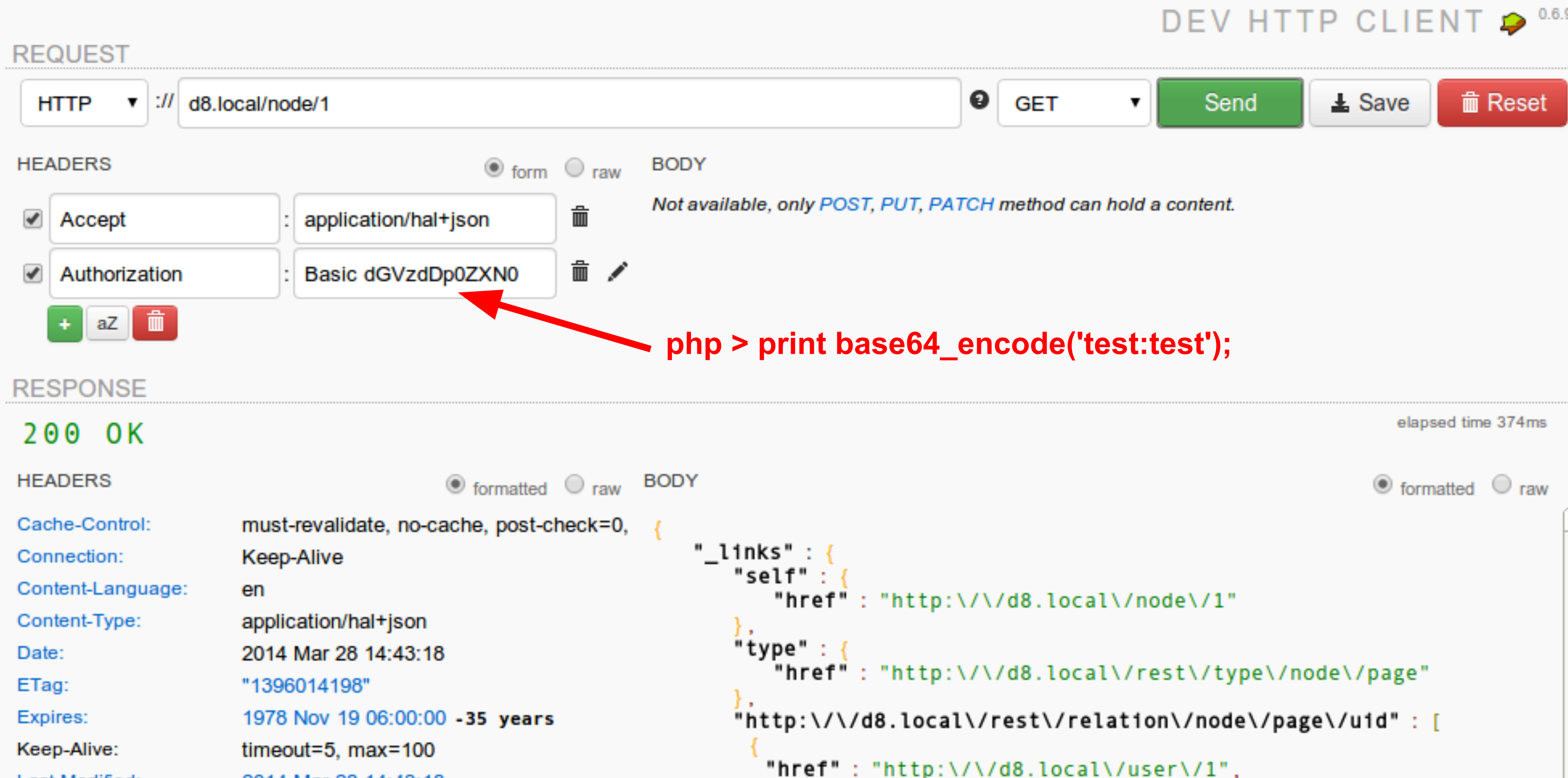
Cookie

Returns authenticated or anonymous user depending on the presence of a cookie.

Basic Auth

Checks if user & password are in the request headers and finds a matching user in the DB.

Basic Auth example



REQUEST

HTTP ▾ :// d8.local/node/1 GET ▾ Send Save Reset

HEADERS ● form ○ raw

☑ Accept : application/hal+json

☑ Authorization : Basic dGVzdDp0ZXN0

+ aZ

BODY

Not available, only POST, PUT, PATCH method can hold a content.

php > print base64_encode('test:test');

RESPONSE

200 OK

elapsed time 374ms

HEADERS ● formatted ○ raw

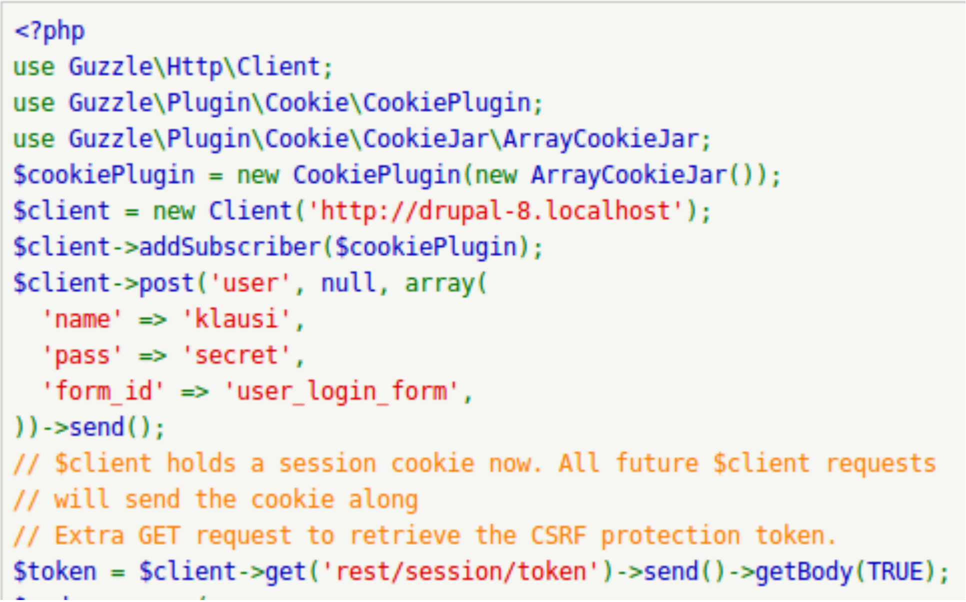
Cache-Control:	must-revalidate, no-cache, post-check=0,
Connection:	Keep-Alive
Content-Language:	en
Content-Type:	application/hal+json
Date:	2014 Mar 28 14:43:18
ETag:	"1396014198"
Expires:	1978 Nov 19 06:00:00 -35 years
Keep-Alive:	timeout=5, max=100

BODY ● formatted ○ raw

```
{
    "_links" : {
        "self" : {
            "href" : "http:\/\/d8.local\/node\/1"
        },
        "type" : {
            "href" : "http:\/\/d8.local\/rest\/type\/node\/page"
        },
        "http:\/\/d8.local\/rest\/relation\/node\/page\/uid" : [
            {
                "href" : "http:\/\/d8.local\/user\/1",
```

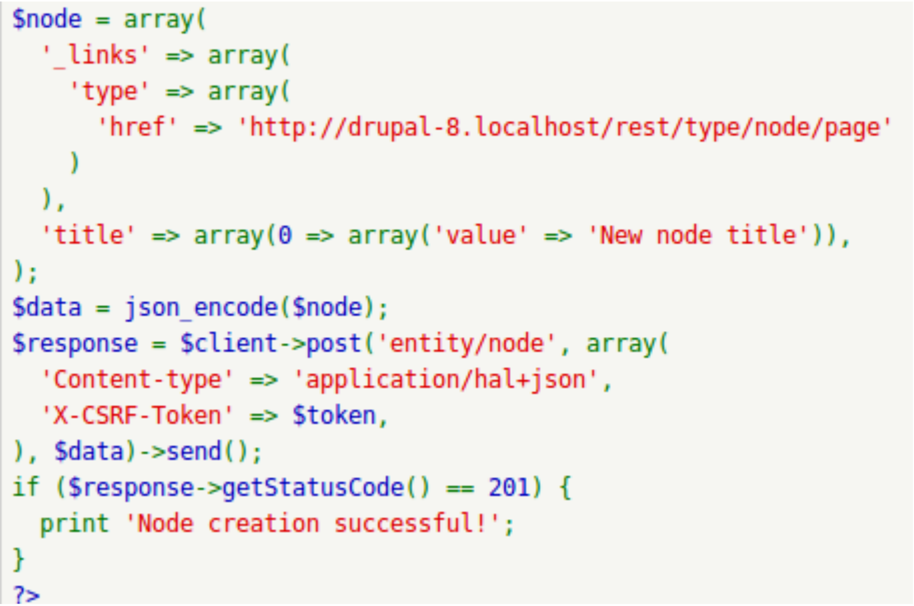
Cookie auth example

1. Obtain a cookie for a Drupal user.



```php
<?php
use Guzzle\Http\Client;
use Guzzle\Plugin\Cookie\CookiePlugin;
use Guzzle\Plugin\Cookie\CookieJar\ArrayCookieJar;
$cookiePlugin = new CookiePlugin(new ArrayCookieJar());
$client = new Client('http://drupal-8.localhost');
$client->addSubscriber($cookiePlugin);
$client->post('user', null, array(
  'name' => 'klausi',
  'pass' => 'secret',
  'form_id' => 'user_login_form',
))->send();
// $client holds a session cookie now. All future $client requests
// will send the cookie along
// Extra GET request to retrieve the CSRF protection token.
$token = $client->get('rest/session/token')->send()->getBody(TRUE);
```

2. Add the cookie id to the request.



```php
$node = array(
  '_links' => array(
    'type' => array(
      'href' => 'http://drupal-8.localhost/rest/type/node/page'
    )
  ),
  'title' => array(0 => array('value' => 'New node title')),
);
$data = json_encode($node);
$response = $client->post('entity/node', array(
  'Content-type' => 'application/hal+json',
  'X-CSRF-Token' => $token,
), $data)->send();
if ($response->getStatusCode() == 201) {
  print 'Node creation successful!';
}
?>
```

https://drupal.org/node/2076725

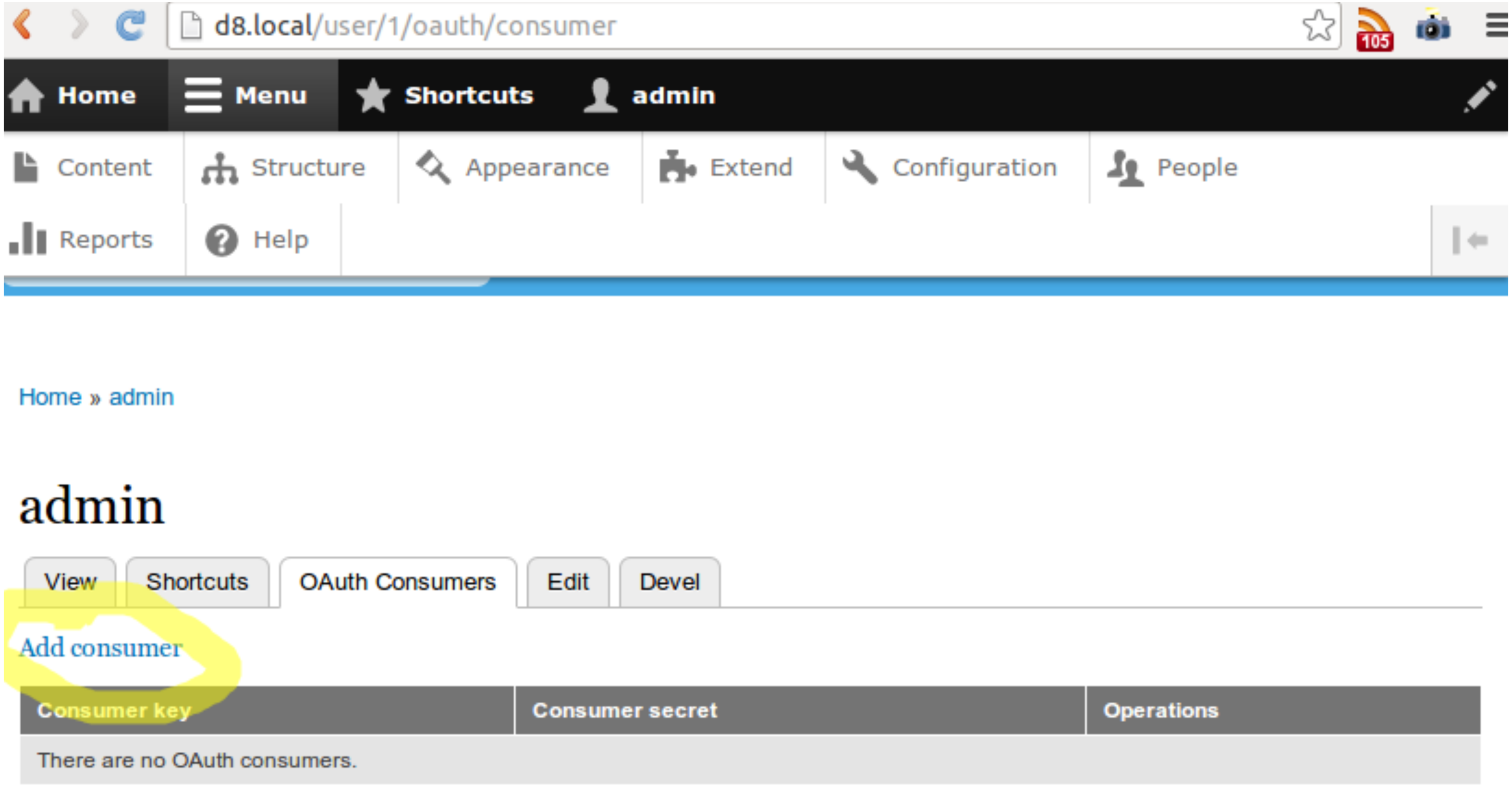
Auth Providers in contrib: OAuth

Supports OAuth 1.0a protocol (Twitter, Flickr).

No support for OAuth2 (Facebook) yet :-(

Will be implemented at OAuth2 Server

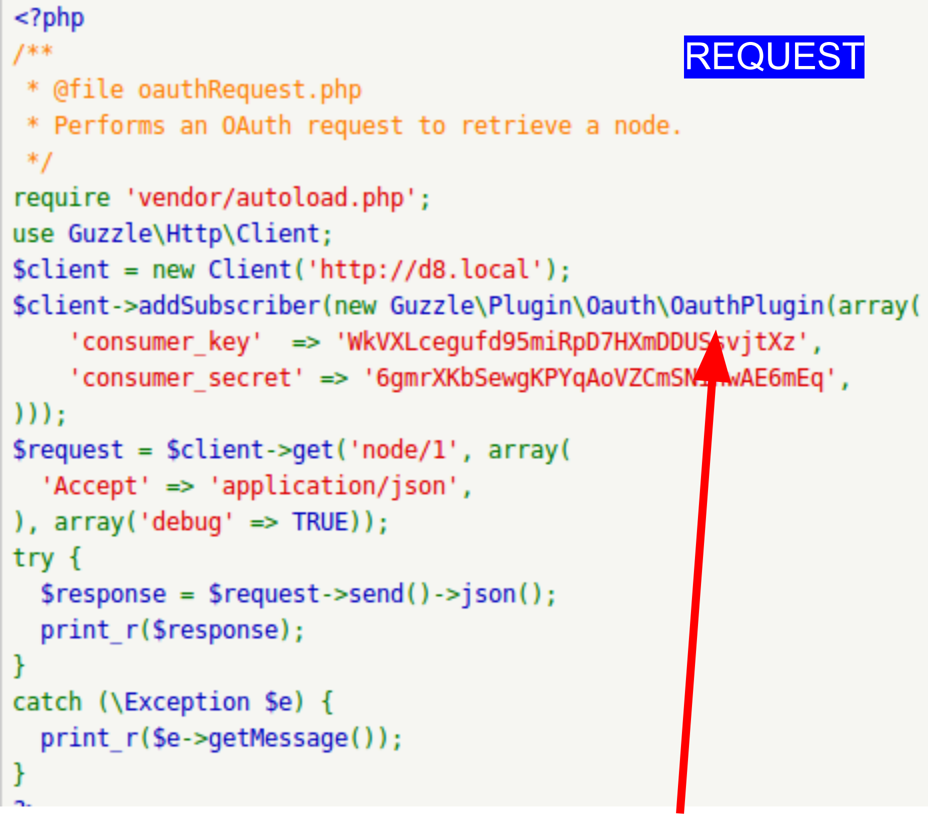
Oauth setup



Consumer key	Consumer secret	Operations
There are no OAuth consumers.		

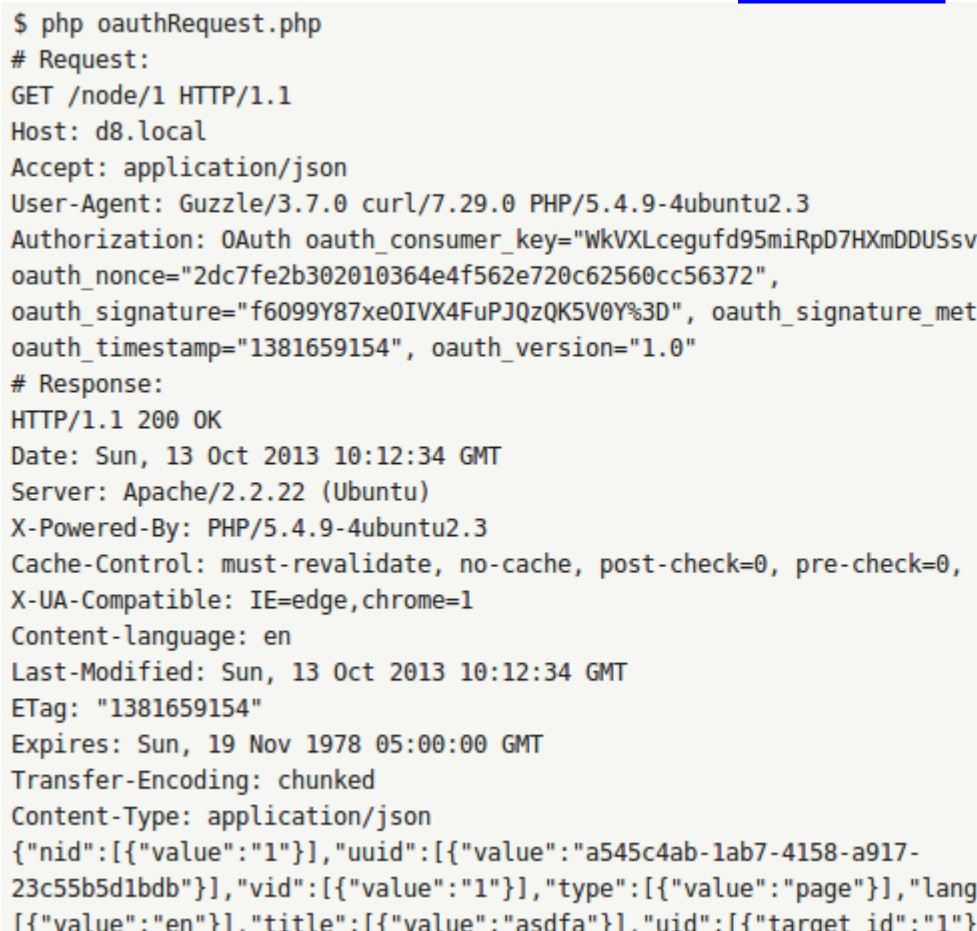
OAuth example request

REQUEST



```php
<?php
/**
 * @file oauthRequest.php
 * Performs an OAuth request to retrieve a node.
 */
require 'vendor/autoload.php';
use Guzzle\Http\Client;
$client = new Client('http://d8.local');
$client->addSubscriber(new Guzzle\Plugin\Oauth\OauthPlugin(array(
    'consumer_key'   => 'WkVXLcegufd95miRpD7HXmDDUSsvjtXz',
    'consumer_secret' => '6gmrXKbSewgKPYqAoVZCmSN[illegible]wAE6mEq',
)));
$request = $client->get('node/1', array(
  'Accept' => 'application/json',
), array('debug' => TRUE));
try {
  $response = $request->send()->json();
  print_r($response);
}
catch (\Exception $e) {
  print_r($e->getMessage());
}
?>
```

RESPONSE



```
$ php oauthRequest.php
# Request:
GET /node/1 HTTP/1.1
Host: d8.local
Accept: application/json
User-Agent: Guzzle/3.7.0 curl/7.29.0 PHP/5.4.9-4ubuntu2.3
Authorization: OAuth oauth_consumer_key="WkVXLcegufd95miRpD7HXmDDUSsv
oauth_nonce="2dc7fe2b302010364e4f562e720c62560cc56372",
oauth_signature="f6O99Y87xeOIVX4FuPJQzQK5V0Y%3D", oauth_signature_met
oauth_timestamp="1381659154", oauth_version="1.0"
# Response:
HTTP/1.1 200 OK
Date: Sun, 13 Oct 2013 10:12:34 GMT
Server: Apache/2.2.22 (Ubuntu)
X-Powered-By: PHP/5.4.9-4ubuntu2.3
Cache-Control: must-revalidate, no-cache, post-check=0, pre-check=0,
X-UA-Compatible: IE=edge,chrome=1
Content-language: en
Last-Modified: Sun, 13 Oct 2013 10:12:34 GMT
ETag: "1381659154"
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Transfer-Encoding: chunked
Content-Type: application/json
{"nid":[{"value":"1"}],"uuid":[{"value":"a545c4ab-1ab7-4158-a917-
23c55b5d1bdb"}],"vid":[{"value":"1"}],"type":[{"value":"page"}],"lang
[{"value":"en"}],"title":[{"value":"asdfa"}],"uid":[{"target_id":"1"}
```

https://drupal.org/project/guzzle_oauth

http://lollaffs.com/wp-content/uploads/2013/03/LOL-Dog-thinking.jpg

¿How does it work?

Client

Request

/latest-news
Authorization: Basic pvcGVuIHNlc2ZQ==

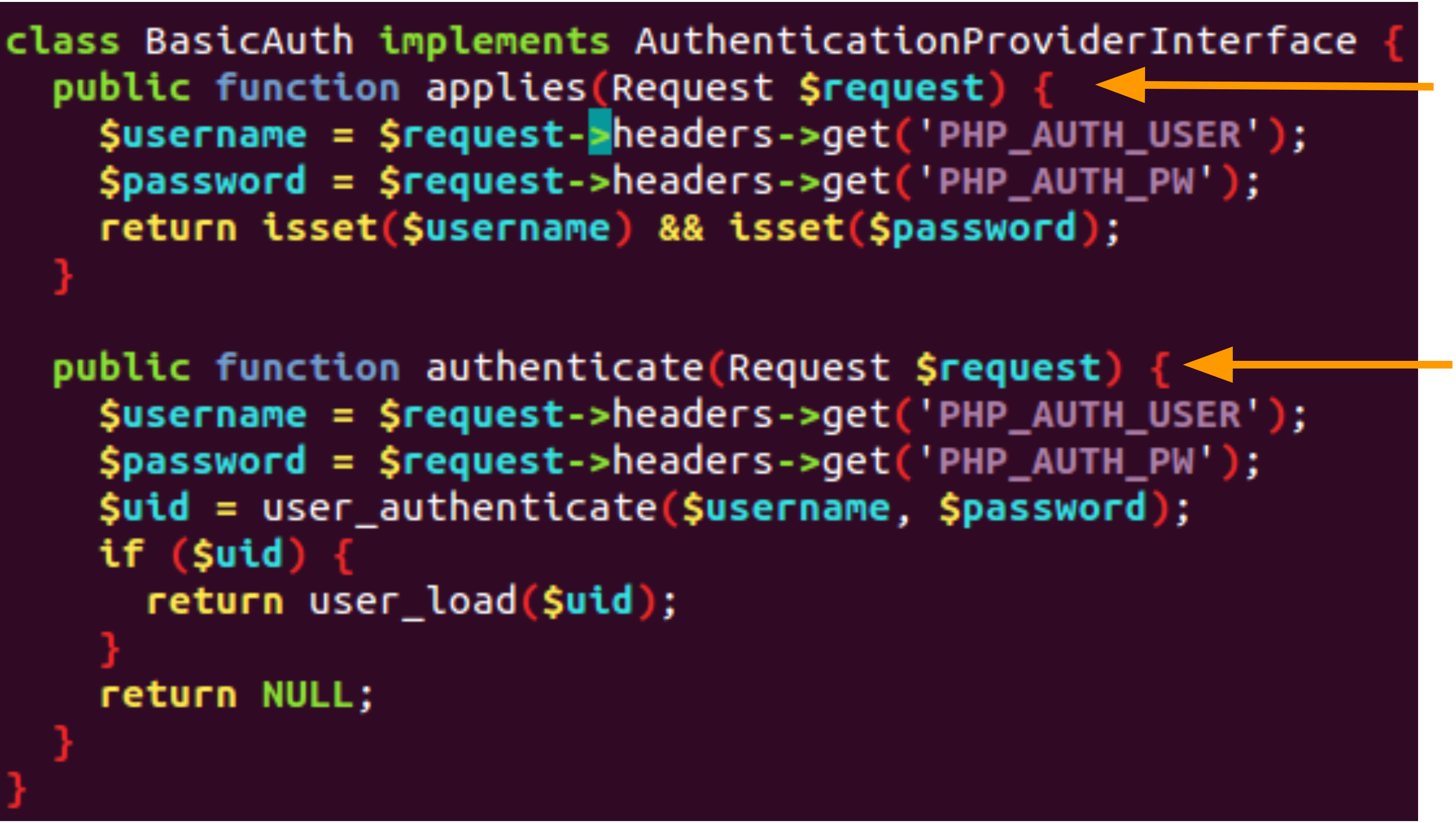
OK 200

- DrupalCamp Spain is a total success
- Geeks in Valencia's Biopark are found sleeping with the gorilas after a fun night
- Álvaro Hurtado disappointed the audience by not doing a striptease

Server

BUT
RAVE
EMERGENCIA
AUTHENTICATION
but
RAVE MARKET
http://aunquedamuchoquedecir.files.wordpress.com/2013/10/entrada-rave-market-madrid-en-sa

Example: Basic Authentication class



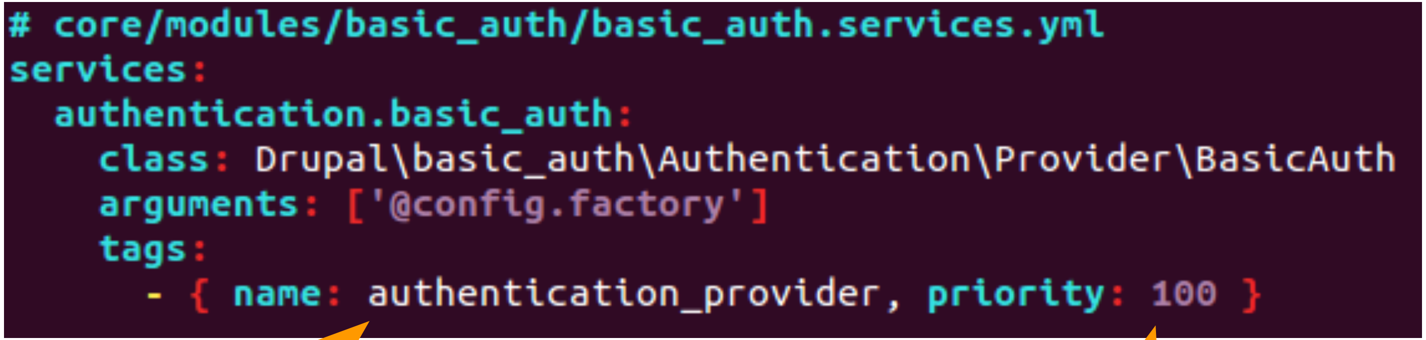
```
class BasicAuth implements AuthenticationProviderInterface {
  public function applies(Request $request) {
    $username = $request->headers->get('PHP_AUTH_USER');
    $password = $request->headers->get('PHP_AUTH_PW');
    return isset($username) && isset($password);
  }

  public function authenticate(Request $request) {
    $username = $request->headers->get('PHP_AUTH_USER');
    $password = $request->headers->get('PHP_AUTH_PW');
    $uid = user_authenticate($username, $password);
    if ($uid) {
      return user_load($uid);
    }
    return NULL;
  }
}
```

Quick check to see if we can authenticate

If the above is TRUE, proceed and attempt to extract a $user.

Basic authentication service

The Authentication Manager looks for services tagged as **authentication_provider**

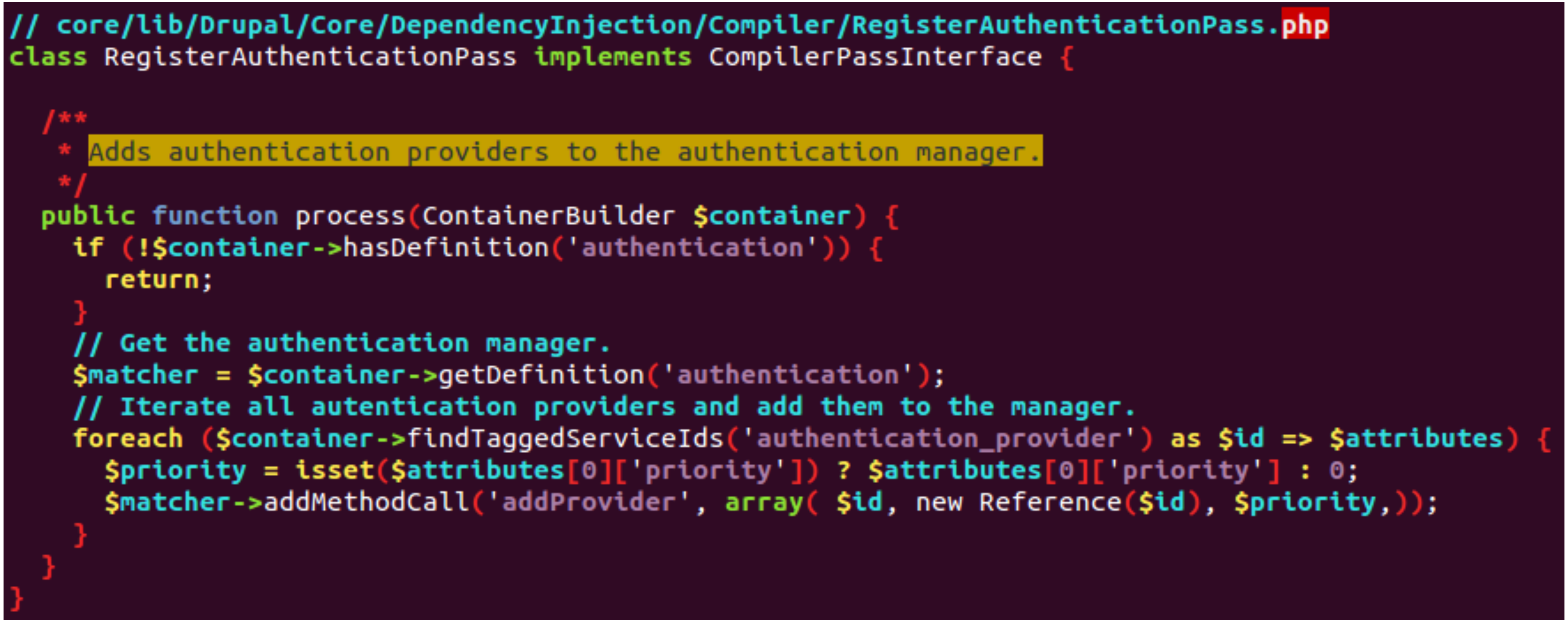


```
# core/modules/basic_auth/basic_auth.services.yml
services:
  authentication.basic_auth:
    class: Drupal\basic_auth\Authentication\Provider\BasicAuth
    arguments: ['@config.factory']
    tags:
      - { name: authentication_provider, priority: 100 }
```

This makes the class discoverable.

Higher priority means that it will
try to authenticate before others

Loading authentication providers



```php
// core/lib/Drupal/Core/DependencyInjection/Compiler/RegisterAuthenticationPass.php
class RegisterAuthenticationPass implements CompilerPassInterface {

  /**
   * Adds authentication providers to the authentication manager.
   */
  public function process(ContainerBuilder $container) {
    if (!$container->hasDefinition('authentication')) {
      return;
    }
    // Get the authentication manager.
    $matcher = $container->getDefinition('authentication');
    // Iterate all autentication providers and add them to the manager.
    foreach ($container->findTaggedServiceIds('authentication_provider') as $id => $attributes) {
      $priority = isset($attributes[0]['priority']) ? $attributes[0]['priority'] : 0;
      $matcher->addMethodCall('addProvider', array( $id, new Reference($id), $priority,));
    }
  }
}
```

Examples

http://hillsidek9academy.com/wp-content/uploads/2013/12/dog-training.jpg

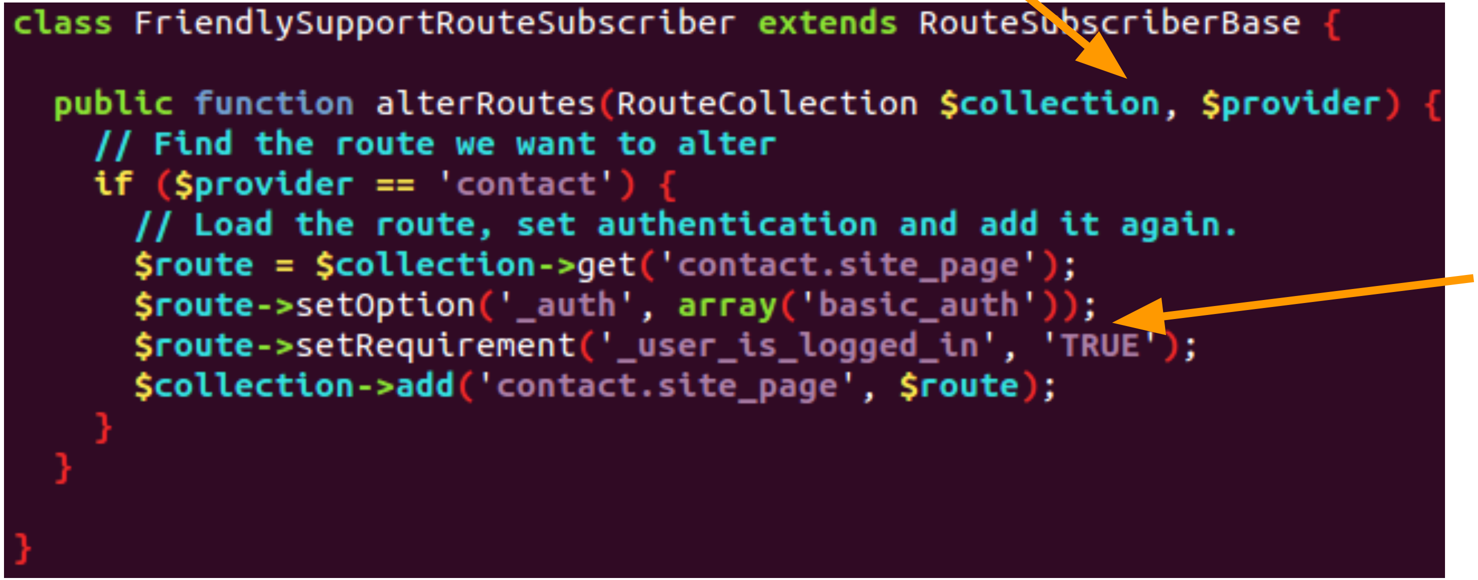
Authenticate an existing route

friendly_support module

Makes it impossible to send support requests by ading HTTP authentication to the Contact form ;D

1. Extend RouteSubscriberBase

$provider is an identifier for a set of routes.
Normally is the module name.

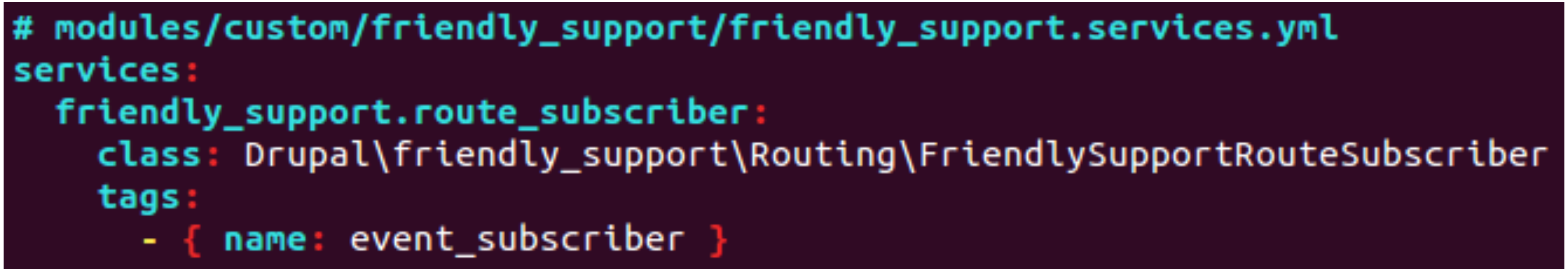


```
class FriendlySupportRouteSubscriber extends RouteSubscriberBase {

  public function alterRoutes(RouteCollection $collection, $provider) {
    // Find the route we want to alter
    if ($provider == 'contact') {
      // Load the route, set authentication and add it again.
      $route = $collection->get('contact.site_page');
      $route->setOption('_auth', array('basic_auth'));
      $route->setRequirement('_user_is_logged_in', 'TRUE');
      $collection->add('contact.site_page', $route);
    }
  }

}
```

Here is where we add authentication rules

2. Make the class a service

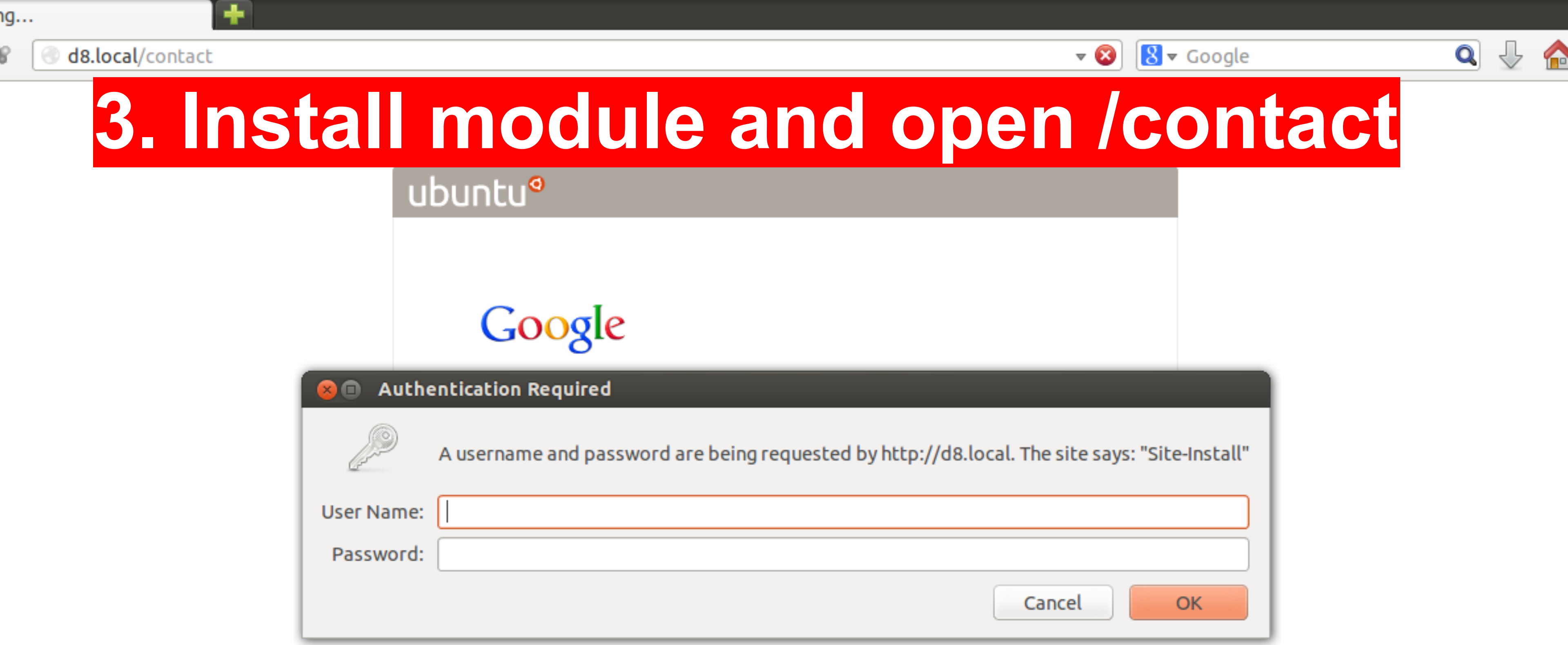
- Just add event_subscriber tag.
- RouteSubscriberBase takes care of the rest.



```yaml
# modules/custom/friendly_support/friendly_support.services.yml
services:
  friendly_support.route_subscriber:
    class: Drupal\friendly_support\Routing\FriendlySupportRouteSubscriber
    tags:
      - { name: event_subscriber }
```

Change record

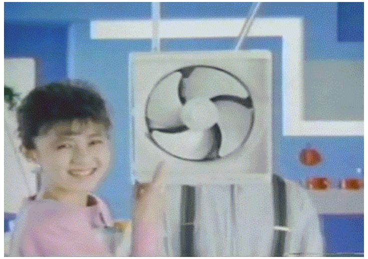
3. Install module and open /contact



A username and password are being requested by http://d8.local. The site says: "Site-Install"

User Name:

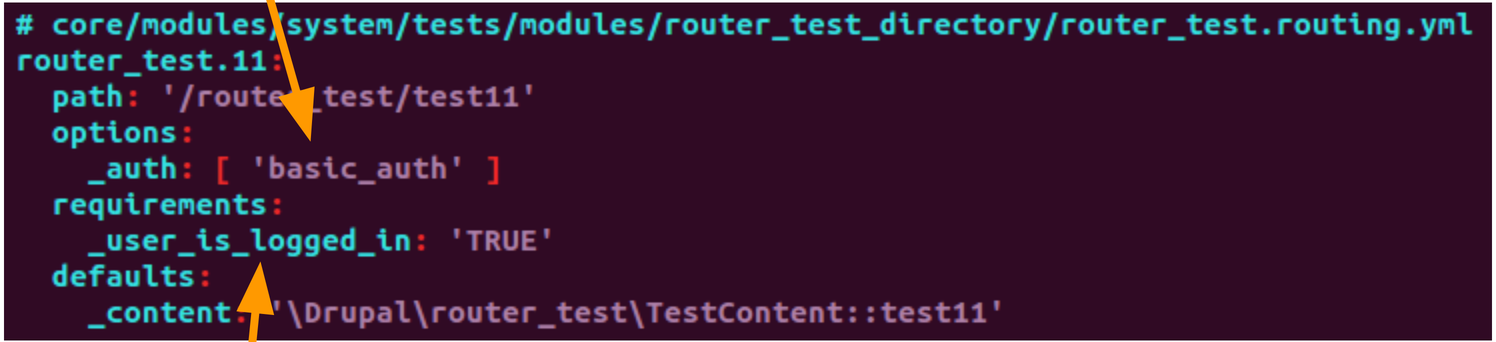
Password:



Authenticate a custom route

We can do it from the route definition.

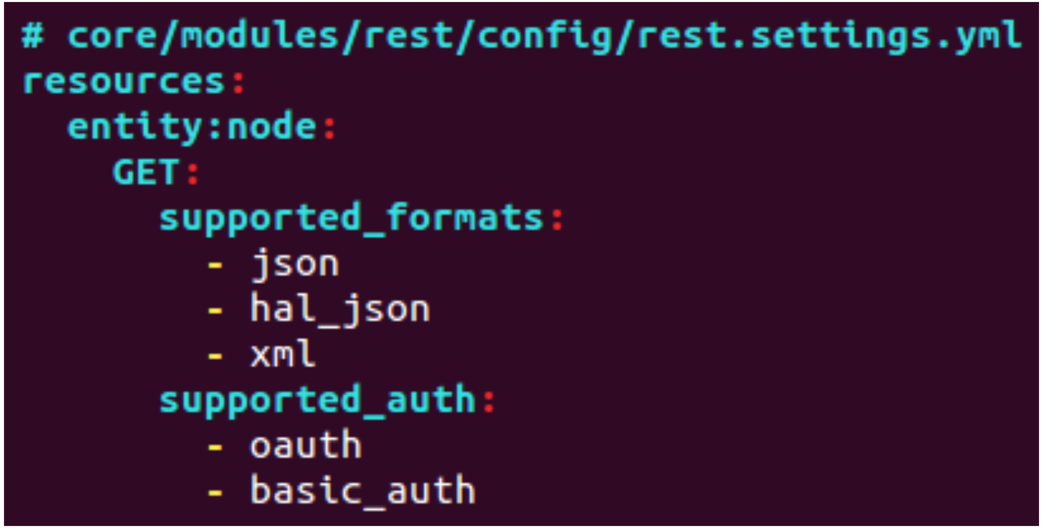
Allowed methods: Basic Authentication



```
# core/modules/system/tests/modules/router_test_directory/router_test.routing.yml
router_test.11:
  path: '/router_test/test11'
  options:
    _auth: [ 'basic_auth' ]
  requirements:
    _user_is_logged_in: 'TRUE'
  defaults:
    _content: '\Drupal\router_test\TestContent::test11'
```

This is part of Authorization: only authenticated users can access.

Authenticate a REST resource

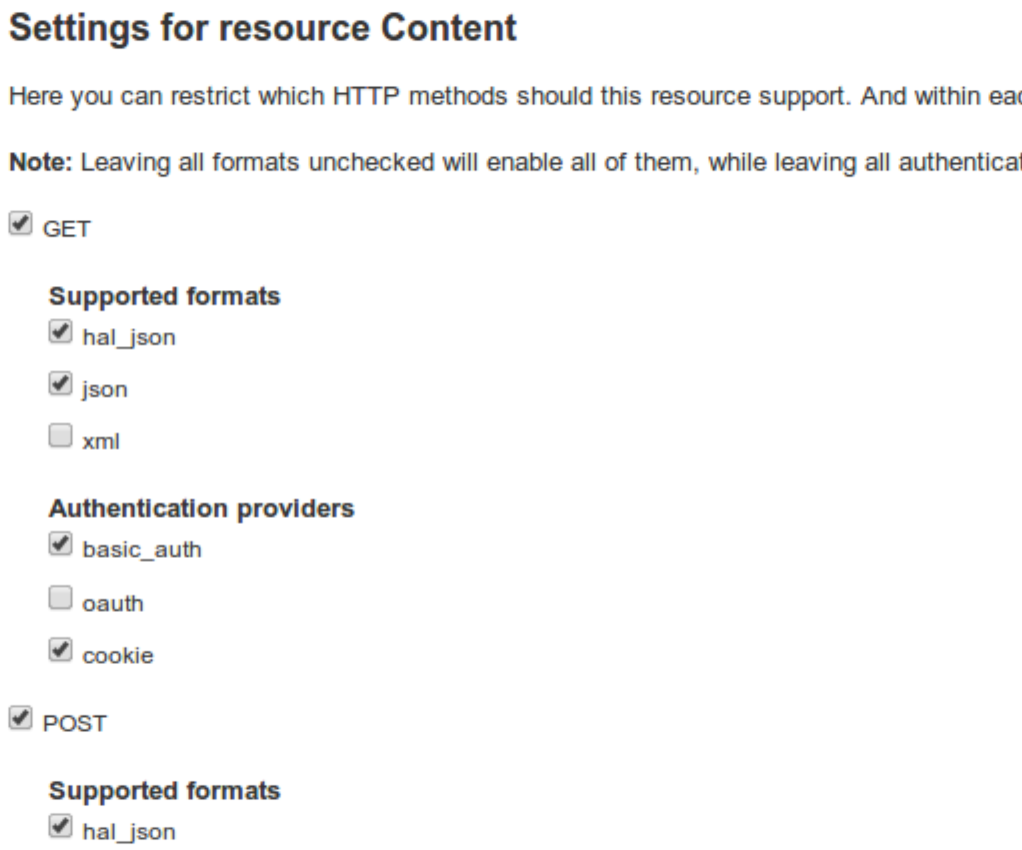


```
# core/modules/rest/config/rest.settings.yml
resources:
  entity:node:
    GET:
      supported_formats:
        - json
        - hal_json
        - xml
      supported_auth:
        - oauth
        - basic_auth
```

Recommended read: REST: exposing data as RESTful web services

REST UI

REST UI offers site builders an interface to set up a REST API, including output formats and authentication.



Settings for resource Content

Here you can restrict which HTTP methods should this resource support. And within eac

Note: Leaving all formats unchecked will enable all of them, while leaving all authenticat

☑ GET

Supported formats

☑ hal_json

☑ json

☐ xml

Authentication providers

☑ basic_auth

☐ oauth

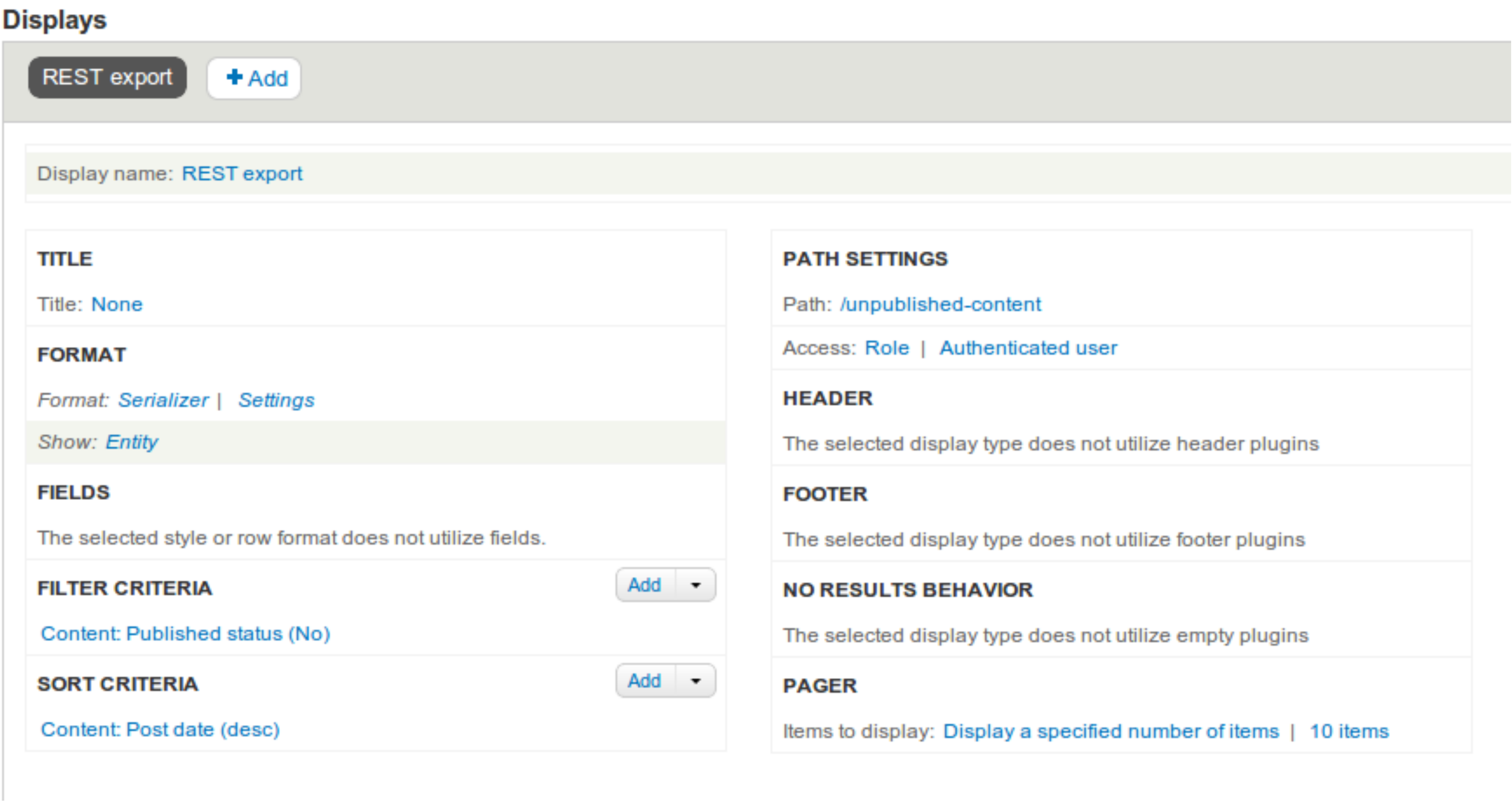
☑ cookie

☑ POST

Supported formats

☑ hal_json

Authenticate a view



Displays

Display name: REST export

TITLE

Title: None

FORMAT

Format: Serializer | Settings

Show: Entity

FIELDS

The selected style or row format does not utilize fields.

FILTER CRITERIA

Content: Published status (No)

SORT CRITERIA

Content: Post date (desc)

PATH SETTINGS

Path: /unpublished-content

Access: Role | Authenticated user

HEADER

The selected display type does not utilize header plugins

FOOTER

The selected display type does not utilize footer plugins

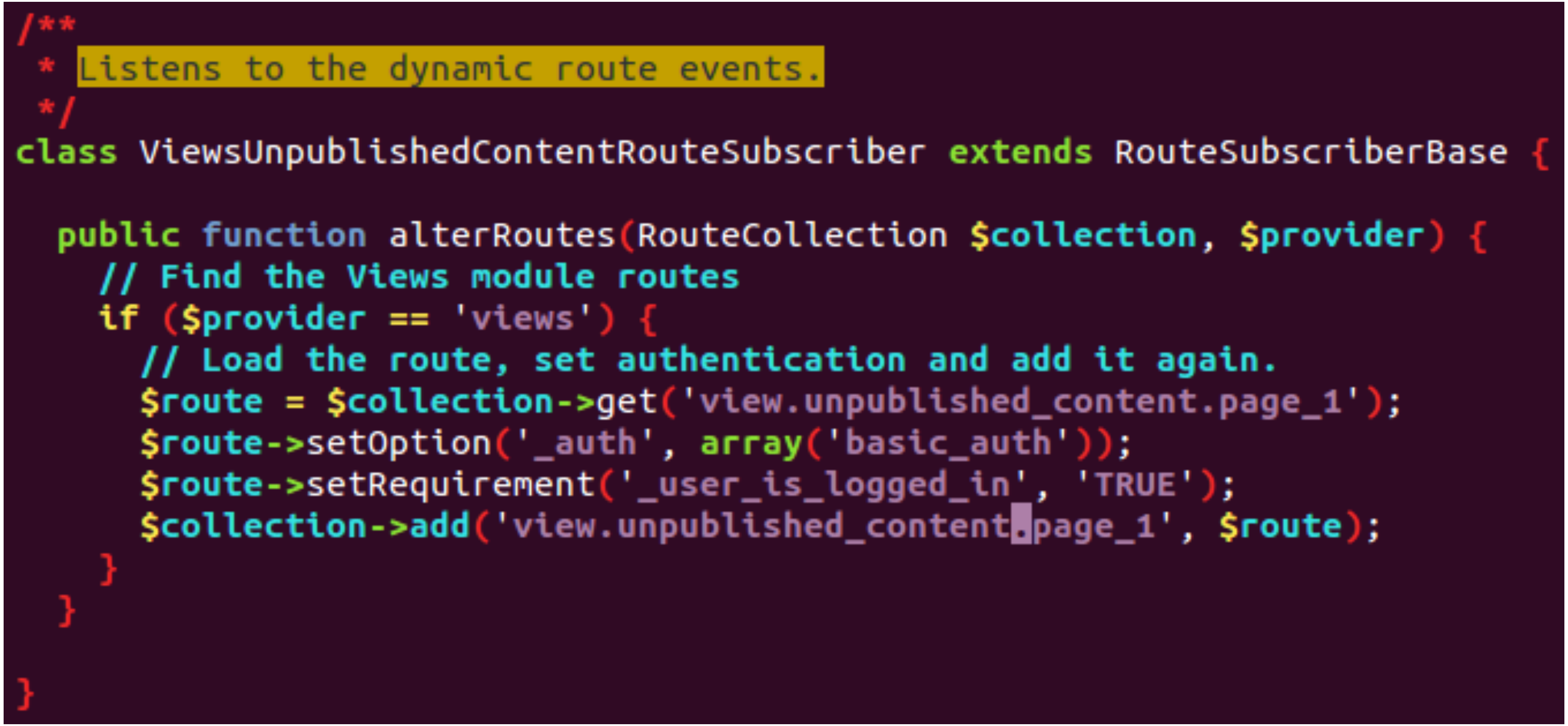
NO RESULTS BEHAVIOR

The selected display type does not utilize empty plugins

PAGER

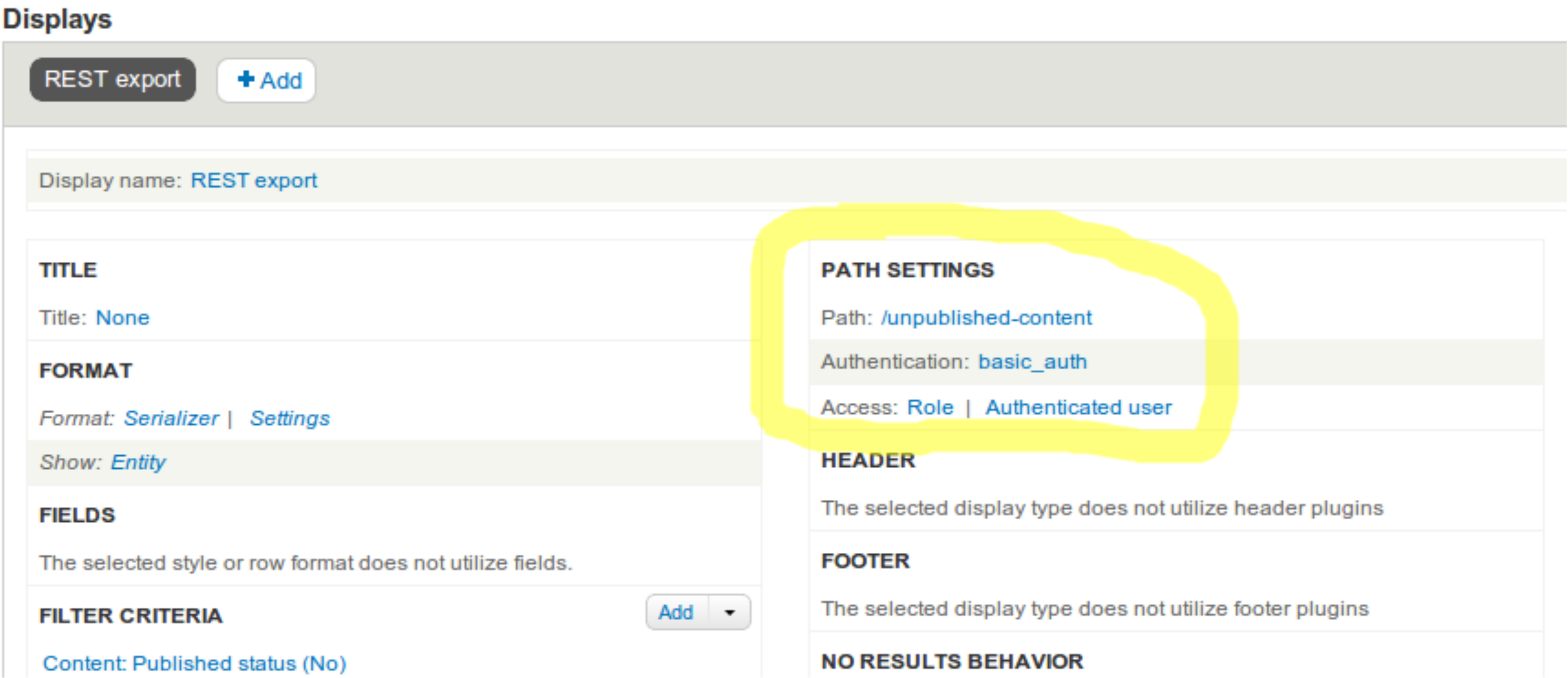
Items to display: Display a specified number of items | 10 items

Authenticate a view trough code



```
/**
 * Listens to the dynamic route events.
 */
class ViewsUnpublishedContentRouteSubscriber extends RouteSubscriberBase {

  public function alterRoutes(RouteCollection $collection, $provider) {
    // Find the Views module routes
    if ($provider == 'views') {
      // Load the route, set authentication and add it again.
      $route = $collection->get('view.unpublished_content.page_1');
      $route->setOption('_auth', array('basic_auth'));
      $route->setRequirement('_user_is_logged_in', 'TRUE');
      $collection->add('view.unpublished_content.page_1', $route);
    }
  }

}
```

Authenticate a view through the UI



Display name: REST export

TITLE

Title: None

FORMAT

Format: Serializer | Settings

Show: Entity

FIELDS

The selected style or row format does not utilize fields.

FILTER CRITERIA

Content: Published status (No)

PATH SETTINGS

Path: /unpublished-content

Authentication: basic_auth

Access: Role | Authenticated user

HEADER

The selected display type does not utilize header plugins

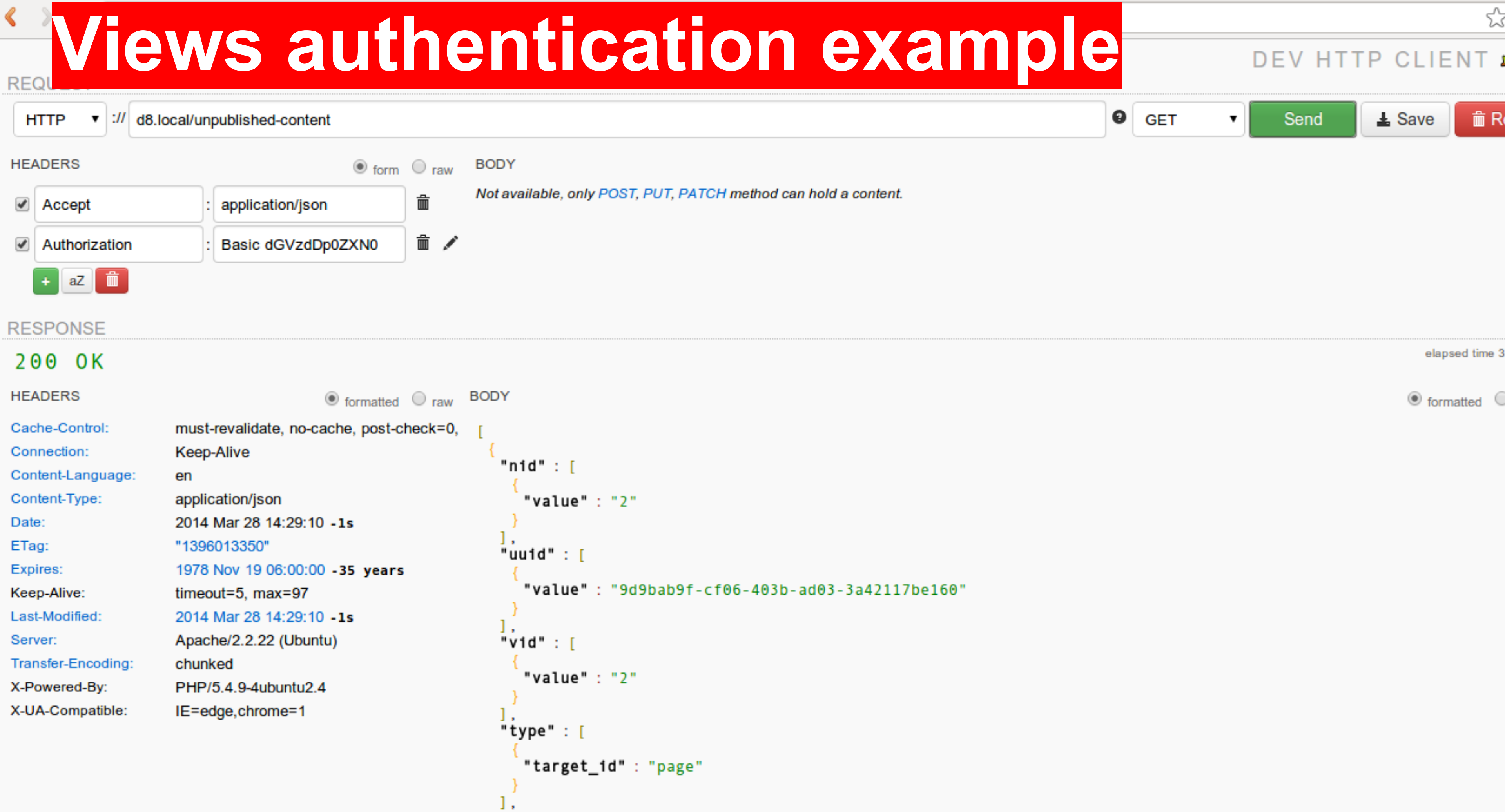
FOOTER

The selected display type does not utilize footer plugins

NO RESULTS BEHAVIOR

https://drupal.org/node/2228141

Views authentication example



HTTP :// d8.local/unpublished-content GET

HEADERS ● form ○ raw

☑	Accept	application/json
☑	Authorization	Basic dGVzdDp0ZXN0

BODY

Not available, only POST, PUT, PATCH method can hold a content.

RESPONSE

200 OK

elapsed time 3

HEADERS ● formatted ○ raw

```
Cache-Control:        must-revalidate, no-cache, post-check=0,
Connection:           Keep-Alive
Content-Language:     en
Content-Type:         application/json
Date:                 2014 Mar 28 14:29:10 -1s
ETag:                 "1396013350"
Expires:              1978 Nov 19 06:00:00 -35 years
Keep-Alive:           timeout=5, max=97
Last-Modified:        2014 Mar 28 14:29:10 -1s
Server:               Apache/2.2.22 (Ubuntu)
Transfer-Encoding:    chunked
X-Powered-By:         PHP/5.4.9-4ubuntu2.4
X-UA-Compatible:      IE=edge,chrome=1
```

BODY ● formatted

```
[
 {
  "nid" : [
    {
      "value" : "2"
    }
  ],
  "uuid" : [
    {
      "value" : "9d9bab9f-cf06-403b-ad03-3a42117be160"
    }
  ],
  "vid" : [
    {
      "value" : "2"
    }
  ],
  "type" : [
    {
      "target_id" : "page"
    }
  ],
```

How to help?

- Add flood support to [OAuth]
- Implement more Auth Providers:
  - OAuth2
  - Digest Authentication
  - IP based authentication

http://urbanchristiannews.com/ucn/DOGS-know-what-were-thinking.jpg

Thanks! Questions?

about.me/juampy

@juampy72