

```

# Elgg htaccess directives
<IfModule mod_headers.c>
    Header set Access-Control-Allow-Origin "*"
    Header set Access-Control-Allow-Credentials: "true"
</IfModule>

<FilesMatch "\.(ttf|otf|woff)$">
    <IfModule mod_headers.c>
        Header set Access-Control-Allow-Origin "*"
    </IfModule>
</FilesMatch>
<Files "htaccess_dist">
    order allow,deny
    deny from all
</Files>

# Don't allow listing directories
Options -Indexes

# Follow symbolic links
Options +FollowSymLinks

# Default handler
DirectoryIndex index.php

#####
# BROWSER CACHING

# The expires module controls the Expires and Cache-Control headers. Elgg sets
# these for dynamically generated files so this is just for static files.
<IfModule mod_expires.c>
    ExpiresActive On
    ExpiresDefault "access plus 1 year"
    Header set Cache-Control "public"
</IfModule>

# Conditional requests are controlled through Last-Modified and ETag headers.
# Elgg sets these on dynamically generated cacheable files so this is just for
# static files. Note: Apache sends Last-Modified by default on static files so
# I don't think we need to be sending ETag for these files.
<FilesMatch "\.(gif|jpeg|png|css|js|pdf|doc|xls|ppt|pptx|pptm|docx|docm|xlsx|xlsx|xlsm|xlt|xltm|xlsb|xltb|xltm|xltm|zip|rar|gz|tar|bz2|swf|flv|wmv|avi|mp4|mp3|ogg|opus|aac|webm|webp|woff|woff2|eot|ttf|otf|woff|woff2)$">

```

```
<FilesMatch \.(jpg|jpeg|gif|png|mp3|flv|mov|avi|ogg|rtm|rtmp|swf|js|css|ico)$ >
```

```
FileETag MTime Size
```

```
</FilesMatch>
```

```
#####
```

```
# PHP SETTINGS
```

```
<IfModule mod_php5.c>
```

```
    # limit the maximum memory consumed by the php script to 1000 MB
```

```
    php_value memory_limit 1000M
```

```
    # register_globals is deprecated as of PHP 5.3.0 - disable it for security reasons.
```

```
    php_value register_globals 0
```

```
    # post_max_size is the maximum size of ALL the data that is POST'ed to php at a time  
(1000 MB)
```

```
    php_value post_max_size 1000M
```

```
    # upload_max_filesize is the maximum size of a single uploaded file (1000 MB)
```

```
    php_value upload_max_filesize 1000M
```

```
    # on development servers, set to 1 to display errors. Set to 0 on production servers.
```

```
    php_value display_errors 0
```

```
</IfModule>
```

```
#####
```

```
# COMPRESSION
```

```
# Turn on mod_gzip if available
```

```
<IfModule mod_gzip.c>
```

```
    mod_gzip_on yes
```

```
    mod_gzip_dechunk yes
```

```
    mod_gzip_keep_workfiles No
```

```
    mod_gzip_minimum_file_size 1000
```

```
    mod_gzip_maximum_file_size 1000000
```

```
    mod_gzip_maximum_inmem_size 1000000
```

```
    mod_gzip_item_include mime ^text/.*
```

```
    mod_gzip_item_include mime ^application/javascript$
```

```
    mod_gzip_item_include mime ^application/x-javascript$
```

```
    # Exclude old browsers and images since IE has trouble with this
```

```
    mod_gzip_item_exclude reqheader "User-Agent: .*Mozilla/4\..*["
```

```
    mod_gzip_item_exclude mime ^image/.*
```

```
</IfModule>
```

```
## Apache2 deflate support if available
```

```
##
```

```
... - - - - -
```

```
## Important note: mod_headers is required for correct functioning across proxies.
##
<IfModule mod_deflate.c>
    AddOutputFilterByType DEFLATE text/html text/plain text/xml text/css text/javascript
application/javascript application/x-javascript
    BrowserMatch ^Mozilla/4 gzip-only-text/html
    BrowserMatch ^Mozilla/4\. [0678] no-gzip
    BrowserMatch \bMSIE !no-gzip

<IfModule mod_headers.c>
    Header append Vary User-Agent env=!dont-vary
</IfModule>

# The following is to disable compression for actions. The reason being is that these
# may offer direct downloads which (since the initial request comes in as text/html and
headers
# get changed in the script) get double compressed and become unusable when downloaded
by IE.
    SetEnvIfNoCase Request_URI action\/* no-gzip dont-vary
    SetEnvIfNoCase Request_URI actions\/* no-gzip dont-vary

</IfModule>

#####
# REWRITE RULES

<IfModule mod_rewrite.c>

RewriteEngine on

# If Elgg is in a subdirectory on your site, you might need to add a RewriteBase line
# containing the path from your site root to elgg's root. e.g. If your site is
# http://example.com/ and Elgg is in http://example.com/sites/elgg/, you might need
#
#RewriteBase /sites/elgg/
#
# here, only without the # in front.
#
# If you're not running Elgg in a subdirectory on your site, but still getting lots
# of 404 errors beyond the front page, you could instead try:
#
```

```
#RewriteBase /
```

```
# If your users receive the message "Sorry, logging in from a different domain is not permitted"
```

```
# you must make sure your login form is served from the same hostname as your site pages.  
# See http://docs.elgg.org/wiki/Login\_token\_mismatch\_error for more info.
```

```
#
```

```
# If you must add RewriteRules to change hostname, add them directly below (above all the others)
```

```
# In for backwards compatibility
```

```
RewriteRule ^pg\([A-Za-z0-9_\-]+\)$ engine/handlers/page_handler.php?handler=$1&%(QUERY_STRING) [L]
```

```
RewriteRule ^pg\([A-Za-z0-9_\-]+\)\(.*\)$ engine/handlers/page_handler.php?handler=$1&page=$2&%(QUERY_STRING) [L]
```

```
RewriteRule ^tag\([.+]\)/?$ engine/handlers/page_handler.php?handler=search&page=$1 [L]
```

```
RewriteRule ^action\([A-Za-z0-9_\-\/]+\)$ engine/handlers/action_handler.php?action=$1&%(QUERY_STRING) [L]
```

```
RewriteRule ^cache\([.]*\)$ engine/handlers/cache_handler.php?request=$1&%(QUERY_STRING) [L]
```

```
RewriteRule ^services/api\([A-Za-z0-9_\-]+\)\(.*\)$ engine/handlers/service_handler.php?handler=$1&request=$2&%(QUERY_STRING) [L]
```

```
RewriteRule ^export\([A-Za-z]+\)\([0-9]+\)/?$ engine/handlers/export_handler.php?view=$1&guid=$2 [L]
```

```
RewriteRule ^export\([A-Za-z]+\)\([0-9]+\)\([A-Za-z]+\)\([A-Za-z0-9_\-]+\)/$ engine/handlers/export_handler.php?view=$1&guid=$2&type=$3&idname=$4 [L]
```

```
RewriteRule xml-rpc.php engine/handlers/xml-rpc_handler.php [L]
```

```
RewriteRule mt/mt-xmlrpc.cgi engine/handlers/xml-rpc_handler.php [L]
```

```
# rule for rewrite module test during install - can be removed after installation
```

```
RewriteRule ^rewrite.php$ install.php [L]
```

```
# Everything else that isn't a file gets routed through the page handler
```

```
RewriteCond %(REQUEST_FILENAME) !-d
```

```
RewriteCond %(REQUEST_FILENAME) !-f
```

```
RewriteRule ^([A-Za-z0-9_\-]+\)$ engine/handlers/page_handler.php?handler=$1 [QSA,L]
```

```
RewriteCond %{REQUEST_FILENAME} !-d
```

```
RewriteCond %{REQUEST_FILENAME} !-f
```

```
RewriteRule ^([A-Za-z0-9_\-\-]+)\/(.*)$ engine/handlers/page_handler.php?handler=$1 &page=$2  
[QSA, L]
```

```
</IfModule>
```