



Autors: **Labi** Raksta ID= **278**

## Mod\_rewrite izmantošana "sadzīvē"

Turpinot iepriekšējā raksta tēmu par mājaslapas pielabināšanu *Google*'i, nolēmu pastāstīt par tādu lietu kā URL (Uniform Resource Locator jeb Vienotais resursu vietrādītis) maskēšanu, izmantojot Apache (apache ir WEB servera programmatūra, sīkāk - [http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server)) moduli **mod\_rewrite**. Godīgi sakot ar URL maskēšanu sīkāk iepazīnos tikai vakar, tāpēc sīki un smalki pastāstīt nesanāks, bet vismaz parādīšu dažus mod\_rewrite izmantošanas piemērus, kurus, iespējams, jau pamanījāt WAP blogu lappusēs.

Kā jau minēju, **mod\_rewrite** ir Apache servera modulis, kas dod iespēju manipulēt ar URL pēc saviem ieskatiem. **Mod\_rewrite** komandas tiek rakstītas Apache lietotāja puses konfigurācijas failā - **.htaccess**. Parasti pirms darba sākšanas pārbauda vai uz servera ir uzstādīts **mod\_rewrite** modulis, visvieglāk to izdarīt, izveidojot nelielu PHP failu - `<?php phpinfo(); ?>` Tālāk veic skripta izpidi uz servera un meklē vai starp visiem PHP parametriem kaut kur ir minēts **mod\_rewrite** (parasti **Loaded modules** sadaļā). Kur un kāpēc izmanto **mod\_rewrite**?

- 1) URL saīsināšanai, piemēram, ja tava raksta adrese ir [wapblogs.eu/ljoti\\_gara\\_adrese.html?vel=kkas](http://wapblogs.eu/ljoti_gara_adrese.html?vel=kkas), tad ar **mod\_rewrite** var panākt, ka šīs lapas adrese būs, piemēram, [wapblogs.eu/01](http://wapblogs.eu/01) - šo URL ir daudz vieglāk atcerēties un tā ir daudz simpātiskāka, vaine?
- 2) lietotāja novirzīšanai uz citu adresi, tas tiek izmantots arī visiem zināmajā adresu īsinātājā <http://ej.uz>
- 3) patiesās lapu/failu adreses slēpšanai, visbiežāk tas ir nepieciešams, lai pasargātu mājaslapu no hakeriem
- 4) lai lapusses adresē būtu atslēgvārdi, kas ir viens no SEO ieteikumiem jaunu lietotāju piesaistīšanai no meklētājiem

Tieši atslēgvārdu ievietošana URL mani pamudināja izmēģināt **mod\_rewrite** darbībā, jo kā jau teicu iepriekšējā rakstā, šobrīd WAP blogi tiek "sapucēti" pēc SEO ieteikumiem cerībā sagaidīt no meklētājiem jaunu lietotāju pieplūdumu. Nav noslēpums, ka *Gūglei* patīk, kad atslēgvārdi atrodas "redzamā vietā", bet visredzamākā vieta, protams, ir mājaslapas URL, to ļoti veiksmīgi izmanto *Vikipēdija*. Pietam ir gadījumi, kad lietotājs meklē atslēgvārdus tikai un vienīgi URL, *Google* ierakstot kaut ko tādu - **inurl:androids**

Ar **mod\_rewrite** man negāja viegli, jo šeit visa pamatā ir regulārie izteicieni (regular expression), kurus es ne pārāk labi pārzinu, taču beigās tiku ar visu galā. Sākšu ar vienkāršu piemēru, lai būtu apmēram skaidrs kā darbojas **mod\_rewrite**. Tātad, ja ir vajadzība pārveidot [domens.lv/realalapa.html](http://domens.lv/realalapa.html) par [domens.lv/sleptalapa](http://domens.lv/sleptalapa), tad var iztikt bez regulāro izteicienu zināšanas, **.htaccess** kods izskatās šādi:

```
RewriteEngine On
RewriteRule ^sleptalapa$ realalapa.html
```

Pirmā rindiņa iedarbina moduli, otrās rindiņas sintakse ir šāda: **RewriteRule url\_paraugs slēptais\_url**



[karogs] sīkāk lasi izmantotajos avotos. Ja sādā veidā nepieciešams noslēpt vairākas lapas, tad katrai lapai jāraksta savs *Rewrite rule* Jāatcerās, ka visas attēlu un CSS failu adreses šajā HTML failā jānorāda sākot no ROOT ("/") mapes, .htaccess jānovieto vienā mapē ar HTML failu. Tagad pārlūkā var ierakstīt [domens.lv/sleptalapa](http://domens.lv/sleptalapa) un pārliecināties, ka viss strādā.

Otrs piemērs, ir `mod_rewrite` izmantošana lietotāja novirzīšanai uz citu adresi. *URL redirect* ir arī WAP blogos, piemēram, mani raksti ir pieejami, ja pārlūkā ieraksta [wapblogs.eu/1](http://wapblogs.eu/1), bet tālāk lietotājs tiek "pārsmešts" uz [wapblogs.eu/blogi.php?kam=1](http://wapblogs.eu/blogi.php?kam=1) (WAP versijā) vai [wapblogs.eu/web/?kam=1](http://wapblogs.eu/web/?kam=1) (WEB versijā). Šajā gadījumā .htaccess saturs izskatās tā:

```
RewriteEngine On
```

```
RewriteRule ^([0-9]+)$ /blogi.php?kam=$1 [R]
```

Redzams, ka šoreiz **url\_paraugs** ir regulārā izteiksme `([0-9]+)`, kas nosaka to, ka URL novirzīšana notiek tajā gadījumā, ja aiz slesha / seko skaitlis, bet **karogs [R]** nozīmē to, ka tiks veikta lietotāja pārmešana uz **slēpto\_url** nevis URL maskēšana.

Pēdējais `mod_rewrite` izmantošanas piemērs WAP blogos ir katra raksta URL maskēšana no [wapblogs.eu/lasit\\_blogu.php?bid=1](http://wapblogs.eu/lasit_blogu.php?bid=1) uz [wapblogs.eu/raksta\\_nosaukums.html](http://wapblogs.eu/raksta_nosaukums.html) Tas ir nedaudz sarežģītāk, jo te ir jāpastrādā ne tikai ar *RewriteRule*, bet arī jāpilnveido skripts. Neko vairāk nestāstīšu, bet ja kādu interesē, tad šajā gadījumā .htaccess saturs ir šāds:

```
RewriteEngine On
```

```
RewriteRule ^(*).html /lasit_blogu.php?raksts=$1&{%QUERY_STRING}
```

Šeit mainīgais `{QUERY_STRING}` ir analogs `$_SERVER['QUERY_STRING']`, tātad .htaccess failā var izmantot tādus mainīgos, kas atrodas PHP masīvā `$_SERVER`

Lai nevajadzētu piņķerēties ar sarežģītām regulārajām izteiksmēm, iesaku izmantot online `mod_rewrite` ģeneratorus - (WEB) <http://generateit.net/mod-rewrite/> un (WEB) [http://webmaster-toolkit.com/mod\\_rewrite-rewriterule-generator.shtml](http://webmaster-toolkit.com/mod_rewrite-rewriterule-generator.shtml) Iznāca garš un murgains raksts, bet cerams, ka galveno domu var saprast.

**Izmantotie avoti:** [http://en.wikipedia.org/wiki/Rewrite\\_engine](http://en.wikipedia.org/wiki/Rewrite_engine), (WEB) [http://workingwith.me.uk/articles/scripting/mod\\_rewrite](http://workingwith.me.uk/articles/scripting/mod_rewrite) un (WEB) <http://easymodrewrite.com>

Uzrakstīts:17:03 21-08-11