

# Apache2: Benutzer je nach Gerät mit mod\_rewrite zur mobilen oder normalen Webseite weiterleiten

[4. Februar 2014](#)

Seitdem die Zahl der Smartphones und Tablets wie dem iPhone, iPad, Android Phones und Tablets, Blackberries, etc. stark angestiegen ist, haben Sie eventuell schon darüber nachgedacht eine mobile Version Ihrer Webseite zu erstellen. Dieses Tutorial erklärt Ihnen, wie Sie Apache so konfigurieren, dass es dem Besucher der Seite die mobile Version anzeigt, wenn er mit einem mobilen Gerät online ist und die normale Seite, wenn er einen Desktop Computer benutzt. Dies werde ich mit Apaches Rewrite Modul machen.

Ich gebe für die Inhalte des Tutorials keinerlei Garantie!

## 1 Vorbemerkung

In diesem Tutorial ist meine "normale" Webseite unter [example.com](#) und [example.com](#) zu erreichen, während meine mobile Webseite [m.example.com](#) heißt. Diese vhosts existieren bereits auf meinem System, ich werde hier nicht erklären, wie man diese erstellt.

## 2 mod\_rewrite aktivieren

Zuerst müssen Sie sicherstellen, dass das Apachemodul mod\_rewrite aktiviert ist. Unter Debian/Ubuntu können Sie es wie folgt aktivieren:

### Quellcode

1. `a2enmod rewrite`

Starten Sie Apache danach neu - unter Debian/Ubuntu tun Sie dies so:

### Quellcode

1. `/etc/init.d/apache2 restart`

## 3 Konfiguration von Apache zur Erlaubnis von Rewrite Rules in den .htaccess Dateien

Meine "normale" Webseite [example.com/example.com](#) hat die vhost Konfigurationsdatei `/etc/apache2/sites-available/www.example.com.vhost` und den Pfad `/var/www/www.example.com/web`.

Meine mobile Seite [m.example.com](#) hat die vhost Konfigurationsdatei `/etc/apache2/sites-available/m.example.com.vhost`

und den Pfad `/var/www/www.example.com/mobile`.

Ich werde die Rewrite Rules für jede Seite in eine .htaccess Datei schreiben (obwohl es auch möglich ist, die Rewrite Rules direkt in die vhost Konfigurationsdatei zu schreiben). Dazu muss ich die vhost Konfigurationen erst so ändern, dass es beiden .htaccess Dateien erlaubt ist, Rewrite Direktive zu beinhalten. Dies mache ich mit der Zeile `AllowOverride All` (welche es den .htaccess

Copyright bleibt bei Teris Cooper und kann jederzeit über [www.root-projekte.de](#) verändert werden.



Dateien erlaubt, alle Einstellungen der vhost Konfiguration zu überschreiben):

### Quellcode

1. vi /etc/apache2/sites-available/www.example.com.vhost

```
[...]  
<Directory /var/www/www.example.com/web/>  
AllowOverride All  
</Directory>  
[...]
```

### Quellcode

1. vi /etc/apache2/sites-available/m.example.com.vhost

```
[...]  
<Directory /var/www/www.example.com/mobile/>  
AllowOverride All  
</Directory>  
[...]
```

Starten Sie Apache danach neu:

### Quellcode

1. /etc/init.d/apache2 restart

## 4 Rewrite Regeln erstellen

Erstellen Sie nun die Rewrite Regeln für die “normale” Webseite [example.com/example.com](http://example.com/example.com), welche alle Benutzer von mobilen Geräten auf die mobile Webseite [m.example.com](http://m.example.com) umleiten werden - Ich beschränke mich hier auf die relevanten Geräte/User Agents, dies sind Android, Blackberry, googlebot-mobile (Googles mobiles Suchbot), IE Mobile, iPad, iPhone, iPod, Opera Mobile, Palm OS und WebOS.

### Quellcode

1. vi /var/www/www.example.com/web/.htaccess

```
<IfModule mod_rewrite.c>  
RewriteEngine On  
RewriteCond %{HTTP_USER_AGENT} "android|blackberry|googlebot-  
mobile|iemobile|ipad|iphone|ipod|opera mobile|palms|webos" [NC]  
RewriteRule ^$ m.example.com/ [L,R=302]  
</IfModule>
```

Für die mobile Webseite [m.example.com](http://m.example.com) sehen die Rewrite Rules, welche alle Besucher, die kein mobiles Gerät benutzen, auf die normale Webseite [example.com/example.com](http://example.com/example.com) umleiten wie folgt aus: - Ich habe nur die RewriteCond Bedingung der eben verwendeten .htaccess Datei negiert:

Copyright bleibt bei Teris Cooper und kann jederzeit über [www.root-projekte.de](http://www.root-projekte.de) verändert werden.



## Quellcode

1. vi /var/www/www.example.com/mobile/.htaccess

```
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteCond %{HTTP_USER_AGENT} "!(android|blackberry|googlebot-
mobile|iemobile|ipad|iphone|ipod|opera mobile|palmos|webos)" [NC]
RewriteRule ^$ example.com/ [L,R=302]
</IfModule>
```

Das war es! Nun können Sie Ihr Setup testen indem Sie beispielsweise m.example.com mit einem normalen Desktop Browser besuchen:

Ist alles richtig eingestellt, sollten Sie zu [example.com](http://example.com) weitergeleitet werden:

Testen Sie es nun andersherum und benutzen ein mobiles Gerät (Ich benutze hier ein Android Phone) um auf [example.com](http://example.com) zuzugreifen:

Sie sollten zu m.example.com weitergeleitet werden:

