La stringa di **query** è la parte dell'URL che segue il simbolo "?". Questa è utilizzata per passare parametri a script CGI o molto più spesso passare parametri a pagine dinamiche.
Nei linguaggi di script lato-server è spesso identificata con la variabile QUERY_STRING, in PHP per esempio è possibile accedere al suo contenuto tramite la variabile di sistema $_SERVER['QUERY_STRING']
.

Manipolare la query tramite le direttive *<Location>*, *Redirect* e *Alias* risulta molto difficile, dato che non è possibile accedere direttamente a quella parte dell'URL, ma tramite il modulo mod_rewrite e le direttive RewriteRule e RewriteCond sarà possibile aggiungere, rimuovere o modificare la stringa di query.
Per informazioni sull'uso delle altre direttive di reindirizzamento leggi l'articolo Aliases, redirecting e rewriting con apache
.

L'analisi della query avviene tramite **RewriteCond** che impone un riscontro con la variabile %{QUERY_STRING}, mentre la modifica dell'URL, compresa la query, viene fatta tramite **RewriteRule**
, se necessario anche usando l'opzione [QSA] per aggiungere altri parametri alla query esistente.

Per capire meglio come gestire la stringa di query, farò qualche esempio di direttive da inserire nei file *htaccess* oppure direttamente nei file di configurazione di Apache. In entrambi i casi è necessario conoscere i fondamentali della configurazione del server, per esempio usando le funzioni *RewriteCond* e *RewriteRule* daremo per scontato che il **mod_rewrite** sia attivo e la direttiva
RewriteEngine
sia impostata ad
"On"
:

RewriteEngine On

Vietare l'accesso ad una o più pagine usando la Query String

```
RewriteCond %{QUERY_STRING} mama     RewriteRule .*/pagina.html - [F]
```

In questo caso verrà negato l'accesso alla pagina http://esempio.com/pagina.html?var=val se la query "
var=val
" contiene la parola "
mama
".

E' importante capire che l'oggetto della *RewriteRule* è il percorso fisico del file richiesto e non l'URL, un esempio potrebbe essere :

"/var/www/miosito/pagina.html" considerando la query l'oggetto diventerebbe "/var/www/miosito/pagina.html?param=mama
".

quindi se volessi bloccare tutte le pagine con la richiesta di un parametro contenente la parola "mama" la direttiva sarebbe:

```
RewriteRule .* - [F]
```

Conoscendo questo meccanismo è possibile creare dei filtri e delle istruzioni più complesse.

Rimuovere la stringa di query

Per rimuovere la query basta aggiungere un punto interrogativo alla fine della regola di Rewrite

```
RewriteCond %{QUERY_STRING} mama  RewriteRule ^test.php /test.php? [L]
```

La query verrà cancellata dalla richiesta ma visivamente l'utente nel suo browser continua a vederla, per nasconderla anche nel browser è necessario un redirect:

```
RewriteRule ^test.php http://esempio.com/test.php? [R,L]
```

Vorrei chiarire che uso l'opzione [L] alla fine del rewrite perché suppongo che sia l'ultima regola da applicare, ma se così non fosse potreste modificarla come preferite.

Aggiungere parametri alla Query String

Nell'esempio successivo imporremo il solito filtro(opzionale) sulla query iniziale e su una determinata pagina(opzionale) e al momento della riscrittura della richiesta verrà aggiunta una copia *parametro/valore*, il tutto usando l'opzione [QSA].

```
RewriteCond %{QUERY_STRING} mama
RewriteRule ^test.php test.php?var=dada [QSA,L]
```

Come gli altri esempi la modifica non è esplicita per l'utente, per cui servirebbe un *redirect* con l'opzione [R].

Ad ogni modo la se la richiesta iniziale era:

http://esempio.com/test.php?param=mama

otterremo

http://esempio.com/test.php?param=mama&var=dada

Riscrittura della richiesta senza modificare la query

Quello che vogliamo è che tenendo conto della query iniziale, quindi imponendo una specifica condizione, deve essere modificato l'URL senza alterare la query.
Questo è molto semplice da farsi, dato che è il comportamento di default.
Quindi data la richiesta :

http://esempio.com/test.php?param=mama

quello che vogliamo ottenere è

http://esempio.com/test2.php?param=mama

La direttiva sarà:

```
RewriteCond %{QUERY_STRING} mama  RewriteRule test.php /test2.php [L]
```

Quindi non specificando alcuna query viene usata l'originale. Faccio sempre presente che fino a quando non viene usata l'opzione di reindirizzamento la richiesta rimane nascosta all'utente, quindi quest'ultimo vedrà sempre *test.php(+query)* ma lo script eseguito sarà *test2.php(+query)* .

Modificare la Query String

Per modificare in generale l'url è possibile usare gli elementi stessi della query iniziale, questo grazie alle *back-references* della direttiva Rewritecond.

Quello che vogliamo è che una richiesta del tipo:

http://esempio.com/test.php?cat=tavolo&item=33&colore=nero

diventi

http://esempio.com/tavolo/item33.php?colore=nero

in pratica vogliamo reindirizzare la richiesta fatta allo script *test.php* verso lo script */tavolo/item33.php* (che deve esistere) mantendo la query sul colore.

```
RewriteCond %{QUERY_STRING} ^cat=(.*)&item=(.*)&colore=(.*)$  RewriteRule test.php /%1/item%2.php?c=%3 [L]
```

Questo reindirizzamento avverrà ogni qualvolta si presenterà una query nella forma specificata nella *RewriteCond*. Le i riferimenti %1, %2 e %3 sono ricavati dall'espressione regolare nella query iniziale e sono utilizzati per creare il nuovo URL.

Le espressioni regolari sulla condizione di riscrittura offrono molti modi per elaborare i parametri della query:

```
RewriteCond %{QUERY_STRING} ^(w+)=(w+)$    RewriteRule ^/path /path/%1/%2?
```

La direttiva entra in azione solo quando il parametro e il suo valore contengono solo lettere e non numeri.

Come sempre in mancanza del reindirizzamento tutta l'operazione è nascosta all'utente.

Creare la query partendo dal percorso

I riferimenti possono essere usati anche nell'espressione regolare della direttiva RewriteRule, ed essenzialmente potremo fare l'opposto di quello che abbiamo fatto nell'esempio precedente, ovvero usare parte del percorso nell'URL per modificare la query.

```
RewriteRule ^([^/]+)/item([^/]+).php /test.php?cat=$1&item=$2 [L]
```

Quindi una richiesta del tipo

```
http://esempio.com/tavolo/item33.php
```

diventerebbe

```
http://esempio.com/test.php?cat=tavolo&item=33
```

Conclusioni

In tutti gli esempi precedenti è possibile utilizzare un reindirizzamento esterno tramite l'opzione [R] nella RewriteRule, ma attenzione che in tal caso è necessario impostare il risultato della regola in forma completa:

```
RewriteRule ^index.php /index.php? [L]
```

diventa

```
RewriteRule ^index.php http://esempio.com/index.php? [R,L]
```

Nel caso è possibile assegnare un codice per il tipo di reindirizzamento, per esempio [R=301] per indicare un redirect permanente.

Questo articolo è diretto a chi come me ha trovato la documentazione ufficiale un po criptica ed ha passato un bel po' di tempo a trovare la giusta configurazione per dei problemi con il web server Apache, spero vivamente che questi esempi possano aiutarvi.

Per ulteriori esempi RewritePathInfo .