

iRule: generic host to uri mapping



Joe Pruitt, 2005-14-10

The [question](#) came in on [DevCentral](#) asking how to dynamically map the company within the host portion of the HTTP request to cause a redirect to a custom uri starting with that company name.



```
I'm in a web hosting environment and I've got a situation where I need to do URL rewrites for the each customer on the BigIP.
```

```
http://www.A.com/ --> http://www.IIS_node.com/A
http://www.B.com/ --> http://www.IIS_node.com/B
.
.
etc (there's a "significant" number of customers... at least 100)
```

One could do a very long if/elseif clause but that wouldn't be any fun now would it? I recommended using a regular expression to strip out the company portion of the uri like the following

```
when HTTP_REQUEST {
  # www.A.com -- domain == A.com, company == A
  regexp {\.([\w]+\)\.com} [HTTP:host] domain company
  if { "" ne $company } {
    HTTP::redirect "http://www.my_vs.com/$company"
  }
}
```

But that could cause unwanted errors if a company is passed in that isn't defined in the backend application. You can get around this by validating the company name with a data group lookup as illustrated below:

```
*** BEGIN STRING DATA GROUP ***
class valid_company_names {
  "A"
  "B"
  "C"
}
*** END STRING DATA GROUP ***

*** BEGIN RULE ***
when HTTP_REQUEST {
  # www.A.com -- domain == A.com, company == A
  regexp {\.([\w]+\)\.com} [HTTP:host] domain company
  if { "" ne $company } {
    if { [matchclass $company equals $::valid_company_names] > 0 } {
      HTTP::redirect "http://www.my_vs.com/$company"
    }
  }
}
}
```

And, finally, if there isn't an exact mapping from domain to uri, but you still want to do the mapping cleanly, you could use a string Data Group that contains the mappings. If the company is found in the Data Group, you can extract the mapping portion of the string with the **findclass** command as follows:

```
*** BEGIN STRING DATA GROUP ***
class valid_company_mappings {
  "A mapping_for_a"
  "B mapping_for_b"
  "C mapping_for_c"
}
*** END STRING DATA GROUP ***

*** BEGIN RULE ***
when HTTP_REQUEST {
```

```
when HTTP_REQUEST {  
  # www.A.com -- domain == A.com, company == A  
  regexp {\.([\w+)\.com} [HTTP:host] domain company  
  if { "" ne $company } {  
  
    # Look for the second string in the data group  
    set mapping [findclass $company $::valid_company_mappings ""]  
    if { "" ne $mapping } {  
      HTTP::redirect "http://www.my_vs.com/$mapping"  
    }  
  }  
}
```

Simple but very powerful iRule if I do say so myself!

See the original forum thread [here](#).

-Joe

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

©2018 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com. Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113