

Winner Winner, iRule Dinner



Colin Walker, 2011-15-02

Last week I [wrote about the iRules challenge issued to the FSEs](#) again, and showed my intended solution. Go take a look at that post for a point of reference if you like, or just dive in.

Today I want to give props where props are due, and show off the winner and two runners up of the FSE iRules Challenge. Big congrats to Harry Kleinbourg, Sudarshan Sivaperumal, and the winner, Karl Vogel. Excellent work, guys! There were many solid entries so the judging was surprisingly close, but these three engineers came the closest to the complete solution while keeping an eye towards efficiency and readability. Really an impressive feat for being so new to the technology.

Okay, enough congratulating, on to the code! First we'll give you the winning iRule, from Karl Vogel.

Winner!

Karl's iRule made proper use of the class command, the getfield command and figured out the trick logging hurdle nicely, rather than getting caught in the intentionally laid trap of trying to log to the mgmt interface directly from the iRule, which is a no go. Another major blunder he avoided was setting the cookie in the request. This was a big one and I knew it would trip up a lot of people (and it did) but Karl handily avoided that pitfall and properly inserted the cookie in the response, as intended. He also showed some stylish HTTP::response formatting that caught my eye, and came very, very close to accomplishing the entire task in his iRule. Karl also made sure that the code was easy to read and maintain, and efficient. Extra points there for sure. There were a few places that could use improvement, and I've offered that feedback as well, but all in all, job very well done! A much deserved win after the time and effort invested, and the product produced.

```
1 when HTTP_REQUEST {
2     ## cookie check - needs work and more QA thought in multiple usage scenarios like different domains etc..
3     if ([HTTP::cookie exists "cname2uri"]) {
4         set my_domain [HTTP::cookie domain "cname2uri"]
5         set my_path [HTTP::cookie path "cname2uri"]
6         HTTP::redirect "http://$my_domain$my_path[HTTP::uri]"
7         log local0. "domain = $my_domain path = $my_path"
8     }
9     ## extract "cname"
10    set cname [string tolower [getfield [HTTP::host] "." 1] ]
11    ## don't run if accessing www.domain.com
12    if ([HTTP::host] != "www.domain.com") {
13        ## check against cname list
14        if { [class match $cname equals my_cname_list] } {
15            HTTP::redirect "http://www.domain.com/$cname[HTTP::uri]"
16        }
17        else {
18            ## catch unknown cnames
19            HTTP::respond 200 content {
20                <html>
21                <head>
22                <title>Whoops!!</title>
23                </head>
24                <body>
25                We are sorry, but the site you are looking for does not exist<br>
26                If you feel you have reached this page in error, please try again.
27                </body>
28                </html>
29            }
30        }
31    }
32    ## need to setup Logging using HSL over eth0:mgmt yet.. might need to look into proxying from syslog_ng or setup route to interface if i can??
33    log local0. "IP [IP::client_addr] originates from [whereis [IP::client_addr] country] and is being redirected to http://www.domain.com/$cname[
34    HTTP::uri] because of cname redirection irule"
35 }
36 when HTTP_RESPONSE {
37     if (![HTTP::cookie exists "cname2uri"]) {
38         ## need to resolve path name problem in cookie
39         HTTP::cookie insert name "cname2uri" value "v1.0" path "/"$cname domain "www.domain.com" }
40 }
```

First Runner Up

In second place we have Sudarshan Sivaperumal, whose iRule was so close to being perfect it was excruciating. Sudarshan was following almost the exact same thought process that I was and at first while reading through his iRule I thought he may have the exact solution I was looking for. He was using classes, logging correctly, responding to the client appropriately and his code was wonderfully compact and neat. The only thing that threw a wrench in the works here was the aforementioned cookie being set in the request and a little formatting around class requests. That being said, this was an extremely solid attempt and the base knowledge of how to solve the problems laid out is definitely there. With a little fine tuning and some more experience working with the products, Sudarshan is going to be an iRuling force to be reckoned with. Well played, sir, and keep at it.

```

1 when HTTP_REQUEST {
2     #verify if repeat user by checking for existence of cookie
3
4     if { [HTTP::cookie exists "f5cname"] } {
5         HTTP::redirect [HTTP::cookie value "f5cname"]
6         set lreason "repeat user: redirecting based on cookie set earlier"
7     } elseif { [findclass [string tolower [HTTP::host]] $::cname_mapping " "] } {
8         #if new user
9         set f5.url [findclass [string tolower [HTTP::host]] $::cname_mapping " "]
10        HTTP::cookie insert name "f5.cname" value $f5.url
11        HTTP::redirect $f5.url
12        set lreason "new user: cookie set"
13    } else {
14        HTTP::respond 200 content "<HTML><HEAD><TITLE>Incorrect URL typed</TITLE></HEAD><BODY>Sorry the URL you typed is incorrect, please check the
same</BODY></HTML>"
15        set lreason "wrong URL typed"
16    }
17    log local0.info "Client: [IP::client_addr] from [whereis [IP::client_addr] country], $lreason"
18 }

```

Second Runner Up

Last but definitely not least, in third place we have Harry Kleinbourg, who put forth a very fine effort in this challenge. Harry showed some out of the box thinking by using the event disable command, had the cookie being inserted in the response, the logging problem was solved...it was a very promising looking rule. The big rub for Harry was that he confused the intent of the challenge and as such ended up re-writing the wrong part of the URL. Confusion will happen and I can't hold that against him too harshly given how well written the code is in general. He's got a bright future in the FSE world if he keeps up this kind of effort in researching and executing solutions to the problems that face those that he's helping. Very strong effort indeed.

```

1 when HTTP_REQUEST {
2     if {[HTTP::cookie exists my_cookie]} {
3         set my_host [HTTP::host]
4         HTTP::header replace Host "[HTTP::cookie my_cookie].[HTTP::host]"
5         pool Appli_HTTP
6         log local0.info "Country of origin: [whereis [IP::client_addr] country] And request: http://$my_host[HTTP::uri]. Has been sent to http://[
HTTP::cookie my_cookie].[HTTP::host][HTTP::uri] because cookie exists"
7         event disable all
8     } else {
9         set my_alias [class match -value [string tolower [HTTP::host]] equals my_list]
10        if {$my_alias == ""} {
11            HTTP::respond 200 content "<HTML><HEAD><TITLE>UNKNOWN CNAME</TITLE></HEAD><BODY>Cname [HTTP::host] IS NOT IN THE LIST</BODY></HTML>"
12            log local0.info "Country of Origin: [whereis [IP::client_addr] country] And request: http://[HTTP::host][HTTP::uri]. Has been bloked because
cname [HTTP::host] is not in the list of allowed cnames"
13        } else {
14            set my_domain [HTTP::host]
15            HTTP::header replace Host "$my_alias.[HTTP::host]"
16            pool Appli_HTTP
17            log local0.info "Country of origin: [whereis [IP::client_addr] country] And request http://$my_domain[HTTP::uri] has been sent to http://$
my_alias.[HTTP::host][HTTP::uri] because alias [HTTP::host] is in the list of allowed cnames"
18        }
19    }
20 }
21
22
23 when HTTP_RESPONSE {
24     if {[info exists my_alias]} {
25         HTTP::cookie insert name "my_cookie" value $my_alias domain $my_domain
26     }
27 }
28

```

Huge congratulations to our winner and finalists. You really deserve a massive amount of props for churning out such solid solutions with so little exposure to the technology. I'm duly impressed, as were others that I've shared the challenge and results with. Keep pushing the ball, digging through [DevCentral](#) and learning the ins and outs of this outstanding tech.

Now that it is all said and done, this iRules challenge has been amazingly fun, educational and rewarding, at least for me. I hope those involved all got something out of it as well, and I'm looking forward to doing it again as long as I didn't mess it up too badly this time. A big thank you to everyone that participated and again to Clint for getting me involved.

#Colin

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

©2016 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