

20 Lines or Less #15



Colin Walker, 2008-08-10

What could you do with your code in 20 Lines or Less? That's the question I ask every week, and every week I go looking to find cool new examples that show just how flexible and powerful iRules can be without getting in over your head.

Finally back into the swing of the 20LoL, I'm happy to give you the 15th edition of this blog series. Today's offering is brought to you by me, F5, and the power of iRules (tm). Okay, not really (tm), but it sounded cool. I still find myself on a journey to seek out the coolest iRule tidbits that I can in hopes of bringing them to you and showing off just how much power you can pack into a minute amount of code in an iRule. This week's examples are nothing less than that, and hopefully you'll find at least one interesting, if not useful.

Advanced URI Rewriting

<http://devcentral.f5.com/Default.aspx?tabid=53&forumid=5&postid=28429&view=topic>

In this example of rewriting a URI we go well past a simple directory change or query rewrite. This example goes a little further into the realm of the possible to show you how to rewrite a UID that is a portion of a complex URI. Thanks to hoolio and the other iRulers that contributed.

```
when RULE_INIT {
    # Set a couple of test query strings
    #set source {a=123&k=456&uid=toto&h=789}
    #set source {uid=toto&a=123&k=456&h=789}
    set source {a=123&k=456&h=789&uid=toto}
    # Split the string into a list on the delimiter &
    log local0. "[split \ $source\ &]: [split $source &]"
    # Create a new query string
    set new_query_string ""
    # Loop through the list and create an array of parameters and
    values
    foreach param_value_pair [split $source &] {
        log local0. "\ $param_value_pair: $param_value_pair"
        # If the current param value pair starts with uid=, then
        prepend it to the list of query string parameters
        if {$param_value_pair starts_with "uid="}{
            set new_query_string ${param_value_pair}${new_query_string}
        } else {
            set new_query_string ${new_query_string}&${param_value_pair}
        }
        log local0. "\ $new_query_string: $new_query_string"
    }
    set new_query_string hxs=1&${new_query_string}
    log local0. "\ $new_query_string: $new_query_string"
}
```

HTTP to HTTPS redirect on 401

<http://devcentral.f5.com/Default.aspx?tabid=53&forumid=5&postid=28235&view=topic>

This iRule is built for a very specific deployment scenario which displays some ... interesting behaviors. The requirement was to redirect back to the proper HTTPS URL for the site if authorization was required. This is done to ensure that things are secure where they need to be before allowing people to enter auth information. I know it's not the most straight-forward way of doing things, but this particular deployment didn't have another workaround, so iRules came to the rescue.

```
when HTTP_REQUEST {
    set host [HTTP::host]
    set uri [HTTP::uri]
}
when HTTP_RESPONSE {
    if {[HTTP::status] == 401}{
        HTTP::redirect "https://$host/$uri"
    }
}
```

Multi-Host HTTP Redirection with Switch

<http://devcentral.f5.com/Default.aspx?tabid=53&forumid=5&postid=28200&view=topic>

This example is a slightly trimmed version of the one provided (by hoolio, yet again) in the forum post. It shows some great ways to use a single switch to match many different domains or partial domains when doing a redirect based on a host. The individual pieces are all pretty straight-forward, but it's a great example of how to build a single, elegant logic flow rather than a bulkier if/else chain.

```
when HTTP_REQUEST {
    log local0. "[IP::client_addr]:[TCP::client_port]: New HTTP
request to [HTTP::host][HTTP::uri]"
    switch -glob [string tolower [HTTP::host]] {
        "*newlifepubs.com" {
            log local0. "[IP::client_addr]:[TCP::client_port]: Matched
1"
            HTTP::redirect "http://www.example.com/nlp"
        }
        "*mpd.example.org" {
            log local0. "[IP::client_addr]:[TCP::client_port]: Matched
2"
            HTTP::redirect "http://staffweb.example.org/mpd/index.aspx"
        }
        "staff.example.org" {
            HTTP::redirect "http://staffweb.example.org/"
        }
        "*movementseverywhere.example.com" {
            HTTP::redirect "http://www.example.org/"
        }
        default {
            log local0. "[IP::client_addr]:[TCP::client_port]: No match"
            discard
        }
    }
}
```

There's another 20LoL for your coding pleasure. Hopefully I've been able to fuel your desire to run out and whip up some awesome iRules yourself. Check back next week for more cool code!

#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