20 Lines or Less #40 – SSL payload searching, user info and ACLs

Colin Walker, 2010-26-08

What could you do with your code in 20 Lines or Less? That's the question I ask (almost) every week for the devcentral community, 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.

This week we dive into parsing SSL encrypted payloads until a given string is found, logging user login info as it comes across the wire, and enforcing a subsite ACL.

http://devcentral.f5.com/Community/GroupDetails/tabid/1082223/asg/50/afv/topic/aft/1172756/aff/5/showtab/groupforums/Default.aspx#1175124

In this first, rather cool, example from user mattrm we get a peek at how he's dealing with logging user info as they log in by making use of the stream profile, STREAM::match command and regular expressions.

```
when STREAM_MATCHED {
# log each match found by the stream filter
log local0. "Stream filter matched:[STREAM::match]"
  set myvar [STREAM::match]
  set 4 "blah"
  regexp {Username=(.+)\sUserpassword=(.+)\sUseremail=(.+)\sUserhomefolder=
(.+)\s} $myvar matched sub1 sub2 sub3
  log local0. "Username=[b64decode $sub1] Userpassword=[b64decode $sub2]
Usermail=[b64decode $sub3]"
}
when LB_SELECTED  {
set serverIP [LB::server addr]
log local0. "LB Server IP $serverIP"
}
```

http://devcentral.f5.com/Community/GroupDetails/tabid/1082223/asg/50/afv/topic/aft/1174268/aff/5/showtab/groupforums/Default.aspx

Bhattman and Chris Miller tag team to answer a thread talking about creating a sub-site ACL and provides this cool little chunk of code. The idea is pretty simple, block access to a specific section of an app unless the client is coming from a specific list of IP addresses. The implementation is wonderfully simple, though, complete with an Access Denied-esque message straight from the iRule.

```
when HTTP_REQUEST {
     if { [class match [string tolower [HTTP::uri]] contains subsite] and !
([[string tolower [HTTP::uri]] contains "/admin/upload") and ![class match
[IP::addr [IP::client_addr]] eq allow] }
       {
            HTTP::respond 200 content "\Acess is forbidden"
        }

}
```

http://devcentral.f5.com/Community/GroupDetails/tabid/1082223/asg/50/afv/topic/aft/1174288/aff/5/showtab/groupforums/Default.aspx

Last but never least, spark rolls up his sleeves and flexes an ounce of his iRuling muscle to show how easy it can be to collect SSL payload data until a given string is found. He even goes one step further to discuss the difference in functionality between the TCP::collect and SSL::collect commands and how the base functionality is similar but not identical. Definitely a cool one.

```
when CLIENTSSL_DATA {
  if { [SSL::payload] contains "the query string" } {
      log local0. "I got the query!"
      SSL::release
  } else {
      SSL::collect
  }
}
```

There you have it, three more examples of iRules coolness in less than 21 lines of code each. See you soon for more iRuling goodness.

#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