Small URL Generator: Part 2

George Watkins, 2010-16-09

As promised, we're back with the second part of our series on the Small URL Generator. Part 1 covered a very basic example that provided conversion of an URL to a shortened bit.ly-style URL. This example works great for creating small URLs based on a pseudo-base 36 encoded key, but it lacked a few of the nifty features that the big providers offer: variable key length and custom short URLs.

Variable Key Length

In part 1, we discussed our motivation for selecting the 6 character static length key. It was short enough to provide valuable URL-shortening capabilities while at the same time it provided enough unique keys (2,176,782,335 to be exact) to ensure adequate storage and minimal probability of a key conflict. This assumption does not take into account the length of URL itself. For instance, a Google Maps URL that may contain 200+ characters would have the same small URL key length as a small URL for www.cnn.com. This does not provide much benefit to shortening www.cnn.com. Now, http://www.cnn.com, which is 19 characters long, will be given a key length of 3 instead of the previous 6. The code can be tweaked to meet your needs, but the defaults should be more than sufficient for most deployments.

URL Length	URL Key Length	Unique Keys
1-19 chars	3 chars	46,655 keys
20-29 chars	4 chars	1,679,615 keys
30-39 chars	5 chars	60,466,175 keys
40+ chars	6 chars	2,176,782,335 keys

Custom Short URLs

The ability to create custom short URLs is one of the cooler features available from URL shortening services. Our first example provided no mechanism for allowing our users to create such custom URLs. In part 2, we added a second text field in the form to allow for a custom assignment. Now http://www.f5.com/products/big-ip/ can be shortened to http://small.url/bigip instead of http://small.url/ll9ma. We love this feature as it eliminates the need to memorize alphanumeric strings as well as makes the short URL more obvious when distributed (assuming someone in the office isn't pulling a prank).

Small URL Generator Sample Code

```
when RULE_INIT {
    set static::small_url_timeout 86400
    set static::small_url_lifetime 86400
    set static::small_url_response_header "<!DOCTYPE HTML PUBLIC \"-//IETF//DTD HTML 2.0//EN\"><html><head> \
        <title>Small URL Generator</title></head><body><center><h1>Small URL Generator</h1> \
        <br>"
    set static::small_url_response_footer "</center></body></html>"

}

when HTTP_REQUEST {
  if { ([HTTP::uri] starts_with "/create?") and ([HTTP::query] ne "") } {
    set url [URI::decode [string tolower [URI::query [HTTP::uri] url]]]
    set custom_url_key [string tolower [URI::query [HTTP::uri] custom_url_key]]
```

```
15:
16:     if { $custom_url_key ne "" } {
17:         if { ([table lookup -subtable small_url $custom_url_key] ne "") } {
18:             HTTP::respond 200 content "$static::small_url_response_header <b><font color=\"ff0000\"> \
19:                 Error: the custom Small URL <a href=\"http://[HTTP::host]/$custom_url_key\"> \
20:                 http://[HTTP::host]/$custom_url_key</a> has already been taken. Please try again. \
21:                 </font></b> $static::small_url_response_footer"
22:         } else {
23:             set url_key $custom_url_key
24:             log local0. "Custom Small URL created for $url with custom key $url_key"
25:         }
26:     } else {
27:         switch -glob [string length $url] {
28:             {[1-9]} { set url_key_length 3 }
29:             {1[0-9]} { set url_key_length 3 }
30:             {2[0-9]} { set url_key_length 4 }
31:             {3[0-9]} { set url_key_length 5 }
32:             default { set url_key_length 6 }
33:         }
34:
35:         set url_key [string tolower [scan [string map {/ "" + ""} [b64encode [md5 $url]]] "%${url_key_length}s"]]
36:     }
37:
38:     if { ([table lookup -subtable small_url $url_key] eq "") } {
39:         table add -subtable small_url $url_key $url $static::small_url_timeout $static::small_url_lifetime
40:         log local0. "Small URL created for $url with key $url_key"
41:     } else {
42:         log local0. "Small URL for $url already exists with key $url_key"
43:     }
44:
45:     HTTP::respond 200 content "$static::small_url_response_header The Small URL for \
46:         <a href=\"$url\">$url</a> is <a href=\"http://[HTTP::host]/$url_key\"> \
47:         http://[HTTP::host]/$url_key</a> $static::small_url_response_footer"
48:   } else {
49:     set url_key [string map {/ ""} [HTTP::path]]
50:     set url [table lookup -subtable small_url $url_key]
51:
52:     if { [string length $url] != 0 } {
53:         log local0. "Found key $url_key, redirecting to $url"
54:         HTTP::redirect $url
55:     } else {
56:         HTTP::respond 200 content "$static::small_url_response_header <form action=\"/create\" \
57:         method=\"get\"><input type=\"text\" name=\"url\">  \
58:         <input type=\"submit\" value=\"make small!\"><br><h4>Make it custom! \
59:         (optional)</h4>http://[HTTP::host]/<input type=\"text\" name=\"custom_url_key\"></form> \
60:         $static::small_url_response_footer"
61:     }
62:   }
63: }
```

Conclusion

iRules began as an HTTP request and response manipulation language, but has matured to the point that we can write complex applications. The addition of tables in BIG-IP version 10 has opened up an entirely new realm of possibilities by which we can store key/value pairs. We hope that the Small URL Generator serves as a testament to the power of iRules and the various ways they can be used in your environments. Stay tuned for more iRule magic and fun tech tips in the near future!

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