

# 20 Lines or Less #81



Jason Rahm, 2014-06-10

*What could you do with your code in 20 Lines or Less?*

That's the question we like to ask from, for, and of (feel free to insert your favorite preposition here) the [DevCentral](#) community, and every time we do, we go looking to find cool new examples that show just how flexible and powerful iRules can be without getting in over your head. Thus was born the 20LoL (20 Lines or Less) series many moons ago. Over the years we've highlighted hundreds of iRules examples, all of which do downright cool things in less than 21 lines of code.

## Shellshock - HTTP Edition

<https://devcentral.f5.com/articles/shellshock-mitigation-with-big-ip-irules>

Joe Pruitt whipped this up as a mitigation for the shellshock bash vulnerability via HTTP. 17 lines of security defenses, at your service!

```
when HTTP_REQUEST {
  set pattern "*( ) \{"*";
  if { [string match $pattern [HTTP::uri]] } {
    log local0. "Detected CVE-2014-6271 attack from '[IP::client_addr]' in URI '[HTTP::uri]'";
    reject;
  } else {
    foreach header_name [HTTP::header names] {
      foreach header_value [HTTP::header values $header_name] {
        if { [string match $pattern $header_value] } {
          log local0. "Detected CVE-2014-6271 attack from '[IP::client_addr]' in HTTP Header $header_";
          reject;
          break;
        }
      }
    }
  }
}
```

## Shellshock - SIP Edition

<https://devcentral.f5.com/articles/shellshock-the-sip-proxy-edition>

Not to be outdone, Nir Zigler contributes his own mitigation for shellshock, though for SIP traffic instead of HTTP. This solution clocks in at 13 lines of code.

```
when CLIENT_DATA {
  set sCVEPattern "*: ( ) \{"*";
  set bCVEFound 0
  if { [string match $sCVEPattern [UDP::payload]] } {
    set bCVEFound 1
  }
}
```

```
}
when SIP_REQUEST {
    if { $bCVEFound } {
        log local0. "Detected CVE-2014-6271 Shellshock attack! IP: '[IP::client_addr]' From: [SIP::fr
        reject
    }
}
}
```

## Insert custom HTTP header determined by VS protocol

<https://devcentral.f5.com/questions/insert-custom-http-header-determined-by-vs-protocol>

And in non-shellshock news, this iRule comes courtesy of superhero Kevin Stewart, who provides an easy solution to informing the backend servers whether the request came via http or https.

```
when CLIENT_ACCEPTED {
    switch [TCP::local_port] {
        "443" {
            set proto "https"
        }
        "80" {
            set proto "http"
        }
    }
}
when HTTP_REQUEST {
    HTTP::header insert VS_HTTP_PROTO $PROTO
}
```

And that's a wrap! Super powered functionality featured in less than 60 lines of code.

---

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | [f5.com](http://f5.com)

F5 Networks, Inc.  
Corporate Headquarters  
[info@f5.com](mailto:info@f5.com)

F5 Networks  
Asia-Pacific  
[apacinfo@f5.com](mailto:apacinfo@f5.com)

F5 Networks Ltd.  
Europe/Middle-East/Africa  
[emeainfo@f5.com](mailto:emeainfo@f5.com)

F5 Networks  
Japan K.K.  
[f5j-info@f5.com](mailto:f5j-info@f5.com)