Categorize SSL traffic by protocol version

Use Case:

With the issues and attacks around SSL protocol layer, it is important that you can get enough details on SSL traffic stats to make efficient decisions.

F5 iRules:

```
when CLIENTSSL_HANDSHAKE {
        ISTATS::incr "ltm.virtual [virtual name] c [SSL::cipher version]" 1
    }
when HTTP_REQUEST {
        if { [string tolower [HTTP::uri]] equals "/sslversions" } {
            set v3 [ISTATS::get "ltm.virtual [virtual name] c SSLv3"]
            set t10 [ISTATS::get "ltm.virtual [virtual name] c TLSv1"]
            set t11 [ISTATS::get "ltm.virtual [virtual name] c TLSv1.1"]
            set t12 [ISTATS::get "ltm.virtual [virtual name] c TLSv1.2"]
            set hbody "<html>\n \
  <head>\n \
    <!--Load the AJAX API-->\n \
    <script type='text/javascript' src='https://www.google.com/jsapi'></script
>\n \
    <script type='text/javascript'>\n \
\n \
      // Load the Visualization API and the piechart package.\n \
      google.load('visualization', '1.0', {'packages':\['corechart'\]});\n \
\n \
      // Set a callback to run when the Google Visualization API is loaded.\n
\
      google.setOnLoadCallback(drawChart);\n \
\n \
      // Callback that creates and populates a data table,\n \
      // instantiates the pie chart, passes in the data and\n \
      // draws it.\n \
```

```
      function drawChart() {\n \
\n \
        // Create the data table.\n \
        var data = new google.visualization.DataTable();\n \
        data.addColumn('string', 'SSL Types');\n \
        data.addColumn('number', 'Versions');\n \
        data.addRows(\[\n \
          \['SSLv3', $v3],\n \
          \['TLSv1', $t10],\n \
          \['TLSv1.1', $t11],\n \
          \['TLSv1.2', $t12]\n \
        \]);\n \
\n \
        // Set chart options\n \
        var options = {'title':'SSL/TLS Versions on [virtual name]',\n \
                       'width':800,\n \
                       'height':600};\n \
\n \
        // Instantiate and draw our chart, passing in some options.\n \
        var chart = new google.visualization.PieChart(document.getElementById(
'chart_div'));\n \
        chart.draw(data, options);\n \
      }\n \
    </script>\n \
  </head>\n \
\n \
  <body>\n \
    <!--Div that will hold the pie chart-->\n \
    <div id='chart_div'></div>\n \
  </body>\n \
</html>\n \
"
            HTTP::respond 200 content $hbody
        }
```

```
    }
```

URL: https://devcentral.f5.com/codeshare/categorize-ssl-traffic-by-version-display-as-graph

NetScaler Solution:

```
add variable sslv2 -type ulong  -init 0
add variable sslv3 -type ulong  -init 0
add variable tlsv1 -type ulong -init 0

add assignment incr_sslv2 -variable $sslv2 -add 1
add assignment incr_sslv3 -variable $sslv3 -add 1
add assignment incr_tlsv1 -variable $tlsv1 -add 1

add rewrite policy sslv2_check CLIENT.SSL.VERSION.EQ(0x002)
incr_sslv2

add rewrite policy sslv3_check CLIENT.SSL.VERSION.EQ(0x300)
incr_sslv3

add rewrite policy tlsv1_check CLIENT.SSL.VERSION.EQ(0x301)
incr_tlsv1
```

Bind these rewrite polices to a SSL type load balancing vserver. These policies are required to collect the stats on incoming traffic and increment the variables created to check for the SSLv2, SSLv3 and TLSv1 protocol versions. Every time the policy gets a new request with one of the protocol versions, it increments the variable to maintain the total count.

```
add responder action sslversions_stats respondwith
q|"<html><head><script type='text/javascript'
src='https://www.google.com/jsapi'></script><script
type='text/javascript'>google.load('visualization', '1.0',
{'packages':['corechart']});" +
"google.setOnLoadCallback(drawChart);function drawChart() {var data
= new google.visualization.DataTable();data.addColumn('string', 'SSL
Types');data.addColumn('number', 'Versions');" +
"data.addRows([\n['SSLv3', "+$sslv3+"],\n['SSLv2',
"+$sslv2+"],\n['TLSv1', "+$tlsv1+"]\n]);var options =
{'title':'SSL/TLS Versions on vServer','width':800,'height':600};" +
"var chart = new
google.visualization.PieChart(document.getElementById('chart_div'));
```

```
chart.draw(data, options);}</script></head><body><div
id='chart_div'></div></body></html>"|


add responder policy pol_check_sslstats
"HTTP.REQ.URL.EQ(\"/sslversions\")" sslversions_stats


Bind the above responder policy to vServer or Global
```

Bind the Responder policy to respective vserver or global bind point. The responder action here is creating a response HTTP page using Javascript and the run time variables to show a chart on the SSL version stats on given vserver.