3.1 – Use 8 lines of code to start a remote connection from a web page

Save a JavaScript file as tutorial1.page.js with the following content:

```
window.onload = function() {
  var gateway = '192.168.12.111', //change this to your Spark gateway address
  server = '192.168.12.117', //change this to your RDP server address
  url = 'ws://' + gateway + '/RDP?server=' + server + '&user=vmuser&pwd=password';
  var r = new svGlobal.Rdp(url);
  r.addSurface(new svGlobal.LocalInterface());
  r.run();
};
```

Save a web page as tutorial1.html with the following content:

```
<!doctype html>
<html>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Spark View (RDP)</title>
  <meta name="viewport" content="width = device-width, initial-scale = 1.0, user-scalable =
yes, minimum-scale = 0.1, maximum-scale = 8" />
  <meta name="apple-mobile-web-app-capable" content="yes" />
  <link rel="stylesheet" href="../hi5.css" />
  <link rel="stylesheet" href="../rdp.css" />
  <script type="text/javascript" src="../appcfg.js"></script>
  <script type="text/javascript" src="../resource.js"></script>
  <script type="text/javascript" src="../hi5core_min.js"></script>
  <script type="text/javascript" src="../hi5_min.js"></script>
  <script type="text/javascript" src="../surface_min.js"></script>
  <script type="text/javascript" src="../rdpcore_min.js"></script>
  <script type="text/javascript" src="../rdp_min.js"></script>
```

```
  <script type="text/javascript" src="tutorial1.page.js"></script>
</head>
<body>
  <div>
    <canvas id="remotectrl"></canvas>
  </div>
</body>
</html>
```

Double click on the html file and open it in your browser. You don’t even need a HTTP server for this demo.

Checklist:

- to enable HTML5.
- <meta http-equiv="X-UA-Compatible" content="IE=edge"> to suggest Internet Explorer using the edge web engine.
- Viewport meta tag for touch interface
- “apple-mobile-web-app-capable” meta tag to allow users to create shortcut on home screen (iOS).
- SparkView JavaScript libraries for RDP connection: appcfg.js (configuration), resource.js (language resource), hi5_min.js (common utilities), surface_min.js (UI), rdp_min.js (RDP decoder). Don’t forget hi5.css and rdp.css style sheets.
- A canvas element with id “remotectrl” to display the remote connection. If you are using a different id, you need to specify it when creating the LocalInterface in your JavaScript code:
  `new svGlobal.LocalInterface(‘MyCanvasId’)`
- Make sure the JavaScript library is loaded before you start a remote connection. A good place is the window.onload event.
- SparkView client will manage the width and height of canvas. Never try to set it up in your code. Never set up the width and height with CSS.
- Make sure you include hi5core_min.js and rdpcore_min.js since 5.0.
- You can set `userWorker: false` in appcfg.js to disable Web Workers (JavaScript threads).

Troubleshooting:

- Check the JavaScript console log from your browser (F12 or Developer Tools)
- Check SparkView log (InstallDir/logs/)
- Internet Explorer may still be in legacy mode.
- You may need to configure hi5.libPath if you are using the JS library in your portal. For example, you can configure this in appcfg.js:
  `hi5.appcfg = {…}; hi5.libPath = ‘pathRelatedToYourWebRoot’;`
- The web worker will load rdpworker_min.js, hi5core_min.js, rdpcore_min.js at runtime. They are supposed to be in the same directory with your web page by default. If not, you can configure hi5.libPath to resolve this issue, but if your gateway is behind VPN and JavaScript rewriter is used by the VPN, this path can be rewritten with a wrong value (VPN

JS rewriter bug). In this case, you can leave hi5.libPath as blank and copy those 3 js files to the web page directory as a workaround.

Revision #2
Created 8 April 2022 10:49:48 by Guest
Updated 8 April 2022 10:54:50 by Guest