ISP Latency

Test Script

Abstract:

(jump to TOC)

This document contains a very simple PowerShell script designed to test and record residential ISP latency so that the amassed data can be given to the ISP as evidence of problems.

Intended Audience:

(jump to TOC)

This document is intended to be used by Windows users who have basic familiarity with PowerShell.

Document Revision and History:

(jump to TOC)

version	date	description
1.0	2021.05.23.1305	document creation

Freeware License and Disclaimer:

(jump to TOC)

This document is freeware, done in the spirit of open-source. You may distribute unchanged copies of this document freely to anyone at anytime. Care has been taken to cite contributing sources and individuals, please do the same. If you find errors in anything contained herein, please comment on them and/or contact me so that we may all help the community.

About the Author:

(jump to TOC)

Daniel L. Benway

Active Directory and Information Security Architect / Engineer
BSc CS, MCSE (NT4, 2000), MCTS (SCCM 2012), CISSP, Security+, Network+, CCNA (2.0), CLP (AD R4)

LinkedIn http://www.LinkedIn.com/in/DanielLBenway

Blog http://www.DanielLBenway.net

twitter @Daniel_L_Benway

Special Thanks:

(jump to TOC)

- Special thanks to my neighbor Linda who helps me badger our ISP when our service is, shall we say, sub-optimal.

Table of Contents:

PowerShell Script:

(jump to TOC)

```
#########################################################################################################################################################################################
# PING_HR_G_Cf.ps1
# PING home router, Google, Cloudflare
#########################################################################################################################################################################################
# Version and Change Log:
# 2020.09.09.1207 - functional version
# 2021.04.10.1948 - put try/catch around all test-connections
#                 - increased home router delay alert threshold to 50ms
#                 - increased default script runtime to 28d
# 2021.04.11.1325 - changed filename from 'proof' to 'ISPProof'
#                 - decreased home router delay alert threshold to 3ms
#                 - added a header line to output files
#                 - changed the format of the output lines
#                 - changed the severity from iconic to numeric
#                 - changed the severity to include 4 levels
#                 - added null handling for all test-connections
# 2021.04.13.0936 - added $errorActionPreference around all test-connections
# 2021.04.16.1101 - cleaned up comments
#                 - moved sleep, and iteration variables to top of script
# 2021.04.16.1203 - added /maxIterations to screen output
#                 - changed audio tones to be longer to accommodate older speakers
# 2021.04.30.1830 - added a user variable for Home Router IP
#                 - added a user variable to toggle sound on/off
#                 - added a user variable for seconds between tests
#                 - added a user variable for maximum days to run
#                 - added logic to check if script is running under Administrator context
#                 - added logic to check for PSh version
#                 - added a severity indicator to the output file lines
#                 - added a days counter to the screen output
# 2021.05.09.1606 - changed the date and time format to use slashes and colons instead of just dots (so Excel is happier)
# 2021.05.20.1601 - changed Cloudfare to Cloudflare
#                 - changed icon to be numeric/symbolic
#                 - changed outfile file header to indicate G or Cf in column headers
#########################################################################################################################################################################################
# This script PINGs Google and Cloudflare with very small amounts of data on a set interval for a chosen duration, so you can amass failure info to give to your ISP.
# The output files are put in c:\temp:
#   the 'full' file contains all results
#   the 'ISPProof' file contains only proof of failure (to give to your ISP)
# If this script cannot PING your home router with no latency, it won't bothter PINGing the remote addresses, and it won't put the failure into the 'ISPProof' file.
# This script will chirp if there is trouble with your home router, and will sound a long tone if there are connectivity problems or latency beyond your home router,
# both while saving all results to the 'full' file.
# Prereqs:
#   your home router is 192.168.1.1 (if not, edit this script)
#   you're running this script in VS Code, using PSh 7 or higher (CLI PSh 7 or higher should work too)
#########################################################################################################################################################################################
# Case martrix:
# ------------------------------------------------------
# Home
# Router    Continue?   Google      Cloudflare   Severity
# ------------------------------------------------------
# fail          n       fail        fail        1
# fail          n       fail        fast        1
# fail          n       fail        slow        1
# fail          n       fast        fail        1
# fail          n       fast        fast        1
# fail          n       fast        slow        1
# fail          n       slow        fail        1
# fail          n       slow        fast        1
# fail          n       slow        slow        1
# ------------------------------------------------------
# fast          y       fail        fail        4
# fast          y       fail        fast        2
# fast          y       fail        slow        3
# fast          y       fast        fail        2
```

```
# fast          y       fast        fast        0
# fast          y       fast        slow        2
# fast          y       slow        fail        3
# fast          y       slow        fast        2
# fast          y       slow        slow        3
# ------------------------------------------------
# slow          n       fail        fail        1
# slow          n       fail        fast        1
# slow          n       fail        slow        1
# slow          n       fast        fail        1
# slow          n       fast        fast        1
# slow          n       fast        slow        1
# slow          n       slow        fail        1
# slow          n       slow        fast        1
# slow          n       slow        slow        1
#################################################################################################################################################################################


#################################################################################################################################################################################
# Setup
#################################################################################################################################################################################
Clear-Host
Set-StrictMode -Version "latest"
$ErrorActionPreference = "inquire"
#-------------------------------------------------------------------------------------------
If ((([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).isInRole([Security.Principal.WindowsBuiltInRole] "Administrator"))
    {Write-Host "INFO: You ARE running this PSh session as an Administrator." -foregroundColor yellow}
Else {Write-Host "INFO: You are NOT running this PSh session as an Administrator." -foregroundColor yellow}
#-------------------------------------------------------------------------------------------
# continue ony if PSh 7 or higher
If ($PSVersionTable.PSVersion.Major -lt 7)
    {
    Write-Host "This script requires PSh 7 or higher." -ForegroundColor Red
    Write-Host ""
    Exit
    } # End If
#-------------------------------------------------------------------------------------------
# continue only if user approves
Write-Host -noNewLine "============== Continue? (Y/N) " -ForegroundColor Yellow
$userResponse = Read-Host
If ($userResponse -ne "Y")
    {
    Write-Host "============== Script exited." -ForegroundColor Yellow
    Write-Host ""
    Exit
    }
#::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
# user-customized variables
$homeRouterIP = "192.168.1.1"
$soundEnabled = $true
$secondsBetweenTests = 15
$maxDaysToRun = 14
#::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
Set-Variable secondsPerDay -option constant -value 86400
$maxIterations = ([Math]::Round(($maxDaysToRun * ($secondsPerDay / $secondsBetweenTests)),1))
#-------------------------------------------------------------------------------------------
# setup output files and paths
If ((Test-Path "C:\Temp") -eq $false) {MkDir "C:\Temp" | Out-Null}
$outFilePath = "C:\temp\PINGData\"
If ((Test-Path $outFilePath) -eq $false) {MkDir $outFilePath | Out-Null}
$now = Get-Date -format "yyyy.MM.dd.HHmm.ss"
    $nowYear = $now.subString(0,4)
    $nowMonth = $now.subString(5,2)
    $nowDay = $now.subString(8,2)
    $nowHour = $now.subString(11,2)
    $nowMinutes = $now.subString(13,2)
    $nowSeconds = $now.subString(16,2)
$nowLine = $nowYear + "/" + $nowMonth + "/" + $nowDay + " " + $nowHour + ":" + $nowMinutes + ":" + $nowSeconds
$fullOutFilePathAndName = $outFilePath + $now + "_fullPINGData.TSV"
```

```
$ISPProofOutFilePathAndName = $outFilePath + $now + "_ISPProofPINGData.TSV"
################################################################################################################################################################################################


################################################################################################################################################################################################
# Functions
################################################################################################################################################################################################
function fun_alert1
    {
    [System.Console]::Beep(150,500) # frequency,duration
    }
################################################################################################################################################################################################
function fun_alert2
    {
    [System.Console]::Beep(150,500) # frequency,duration
    [System.Console]::Beep(150,500) # frequency,duration
    }
################################################################################################################################################################################################
function fun_alert3
    {
    [System.Console]::Beep(150,500) # frequency,duration
    [System.Console]::Beep(150,500) # frequency,duration
    [System.Console]::Beep(150,500) # frequency,duration
    }
################################################################################################################################################################################################
function fun_alert4
    {
    [System.Console]::Beep(150,1500) # frequency,duration
    }
################################################################################################################################################################################################


################################################################################################################################################################################################
# Main Script Block
################################################################################################################################################################################################
$statusLine = "Date and Time" + [char]09 + "Severity" + [char]09 + "Icon" + [char]09 + [char]09 +"Home Router" + [char]09 + "HR Latency" + [char]09 + "HR Status" + [char]09 + [char]09
+ "Google" + [char]09 + "G Latency" + [char]09 + "G Status" + [char]09 + [char]09 + "Cloudflare" + [char]09 + "Cf Latency" + [char]09 + "Cf Status" + [char]09 + [char]09 + "Summary"
# DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
$statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
$statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
$iterationCounter = 0
Do  {
    $iterationCounter ++
    $statusLine = "no status"
    $PINGHomeRouter = $null
    $PINGHomeRouterException = $null
    $PINGGoogle = $null
    $PINGGoogleException = $null
    $PINGCloudflare = $null
    $PINGCloudflareException = $null
    $now = Get-Date -format "yyyy.MM.dd.HHmm.ss"
        $nowYear = $now.subString(0,4)
        $nowMonth = $now.subString(5,2)
        $nowDay = $now.subString(8,2)
        $nowHour = $now.subString(11,2)
        $nowMinutes = $now.subString(13,2)
        $nowSeconds = $now.subString(16,2)
    $nowLine = $nowYear + "/" + $nowMonth + "/" + $nowDay + " " + $nowHour + ":" + $nowMinutes + ":" + $nowSeconds
    #-------------------------------------------------------------------------
    $ErrorActionPreference = "continue"
    Try { $PINGHomeRouter = test-connection $homeRouterIP -count 1 -bufferSize 1 }
    Catch { $PINGHomeRouterException = $_ } # immediately capture the contents of the PowerShell pipe for whatever caused the 'Catch'
    $ErrorActionPreference = "inquire"
    #-------------------------------------------------------------------------
    If ($PINGHomeRouter -eq $null)
    ##############################################################################################################################################################################################
    ########## (HR null) #########################################################################################################################################################################
    ##############################################################################################################################################################################################
        {
```

```
        # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
        $statusLine = $nowLine + [char]09 + "1" + [char]09 + "1    " + [char]09 + [char]09 + "HR:" + [char]09 + "null" + [char]09 + "null" + [char]09 + [char]09 + "G:" + [char]09 +
"untested" + [char]09 + "untested" + [char]09 + [char]09 + "Cf:" + [char]09 + "untested" + [char]09 + [char]09 + "untested" + [char]09 + "home router null"
        $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
        If ($soundEnabled) {fun_alert1}
        }
    ElseIf (($PINGHomeRouter.status -ne "Success") -Or ($PINGHomeRouter.latency -gt 10))
    ##########################################################################################################################################################################
    ########## (HR fail OR HR slow) ##########################################################################################################################################
    ##########################################################################################################################################################################
        {
        # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
        $statusLine = $nowLine + [char]09 + "1" + [char]09 + "1    " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status + [char]09 +
[char]09 + "G:" + [char]09 + "untested" + [char]09 + "untested" + [char]09 + [char]09 + "Cf:" + [char]09 + "untested" + [char]09 + "untested" + [char]09 + [char]09 + "home router no
connectivity, or with latency"
        $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
        If ($soundEnabled) {fun_alert1}
        }
    Else
    ##########################################################################################################################################################################
    ########## (HR pass AND HR fast) #########################################################################################################################################
    ##########################################################################################################################################################################
        {
        $now = Get-Date -format "yyyy.MM.dd.HHmm.ss"
            $nowYear = $now.subString(0,4)
            $nowMonth = $now.subString(5,2)
            $nowDay = $now.subString(8,2)
            $nowHour = $now.subString(11,2)
            $nowMinutes = $now.subString(13,2)
            $nowSeconds = $now.subString(16,2)
        $nowLine = $nowYear + "/" + $nowMonth + "/" + $nowDay + " " + $nowHour + ":" + $nowMinutes + ":" + $nowSeconds
        #-------------------------------------------------------------------------
        $ErrorActionPreference = "continue"
        Try { $PINGGoogle = test-connection 8.8.8.8 -count 1 -bufferSize 1 }
        Catch { $PINGGoogleException = $_ } # immediately capture the contents of the PowerShell pipe for whatever caused the 'Catch'
        Try { $PINGCloudflare = test-connection 1.1.1.1 -count 1 -bufferSize 1 }
        Catch { $PINGCloudflareException = $_ } # immediately capture the contents of the PowerShell pipe for whatever caused the 'Catch'
        $ErrorActionPreference = "inquire"
        #-------------------------------------------------------------------------
        If (($PINGGoogle -eq $null) -Or ($PINGCloudflare -eq $null))
        ##########################################################################################################################################################################
        ########## (HR pass AND HR fast) AND (Google null IOR Cloudflare null) ###################################################################################################
        ##########################################################################################################################################################################
            {
            If (($PINGGoogle -eq $null) -And ($PINGCloudflare -eq $null))
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summaryy
                $statusLine = $nowLine + [char]09 + "1" + [char]09 + "1   " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + "null" + [char]09 + "null" + [char]09 + [char]09 + "Cf:" + [char]09 + "null" + [char]09 + "null" + [char]09 + [char]09 + "Google and Cloudflare
null"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert1}
                }
            #-----------------------------------------------------------------------------------------------
            ElseIf (($PINGGoogle -eq $null) -And ($PINGCloudflare -ne $null))
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summaryy
                $statusLine = $nowLine + [char]09 + "1" + [char]09 + "1    " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + "null" + [char]09 + "null" + [char]09 + [char]09 + "Cf:" + [char]09 + "not null" + [char]09 + "not null" + [char]09 + [char]09 + "Google null,
Cloudflare not null"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert1}
                }
            #-----------------------------------------------------------------------------------------------
            Else # (($PINGGoogle -ne $null) -And ($PINGCloudflare -eq $null))
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summaryy
```

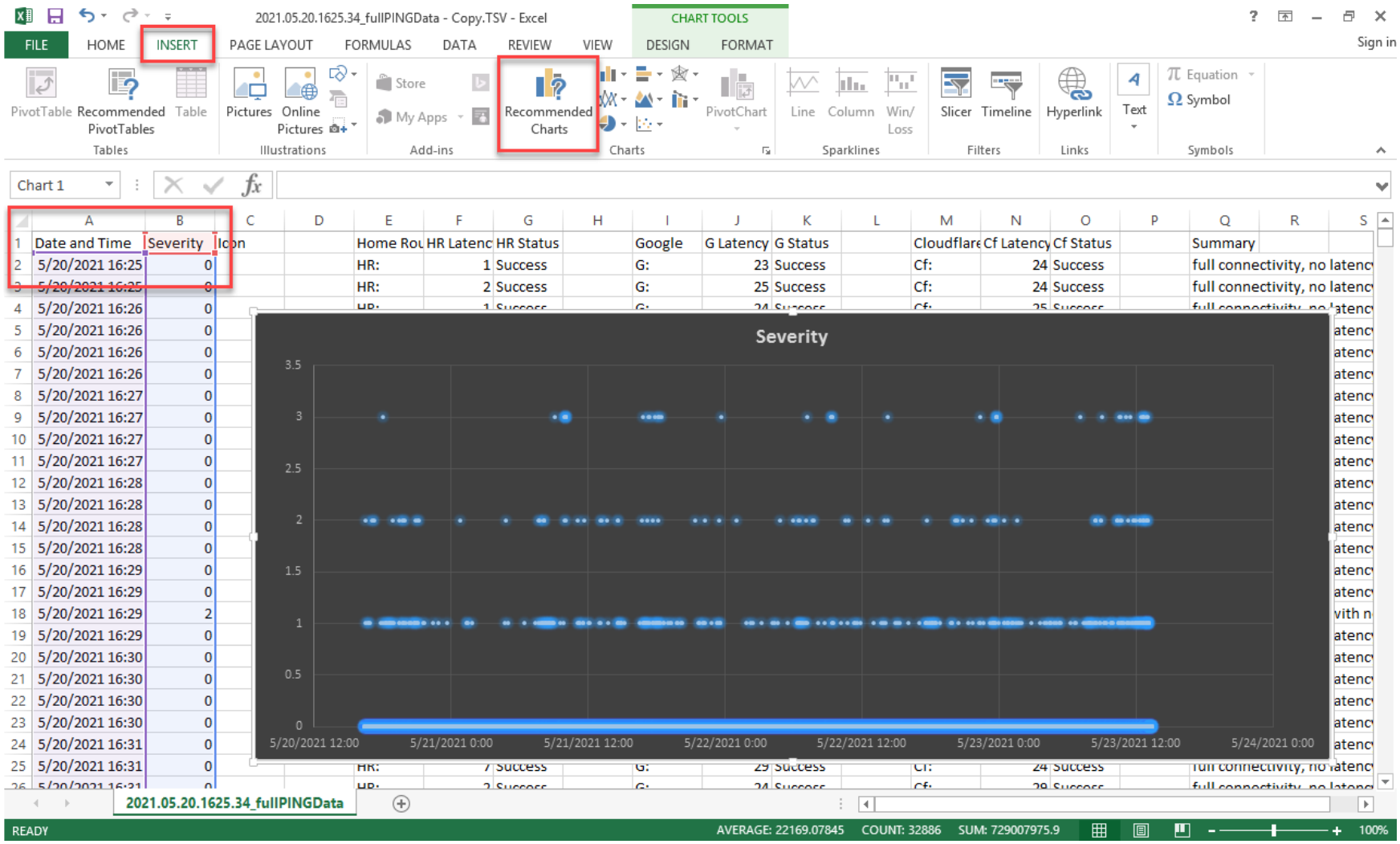
```
                $statusLine = $nowLine + [char]09 + "1" + [char]09 + "1    " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + "not null" + [char]09 + "not null" + [char]09 + [char]09 + "Cf:" + [char]09 + "null" + [char]09 + "null" + [char]09 + [char]09 + "Google not
null, Cloudflare null"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert1}
                }
            }
        ElseIf (($PINGGoogle.status -eq "Success") -And ($PINGCloudflare.status -eq "Success"))
        ##########################################################################################################################################################
        ######### (HR pass AND HR fast) AND (G pass AND Cf pass) ##################################################################################################
        ##########################################################################################################################################################
            {
            If (($PINGGoogle.latency -gt 50) -And ($PINGCloudflare.latency -gt 50))
            ######################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G pass AND Cf pass) AND (G slow AND Cf slow) ####################################################################
            ######################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summaryy
                $statusLine = $nowLine + [char]09 + "3" + [char]09 + "333 " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "full connectivity, overall latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert3}
                }
            ElseIf (($PINGGoogle.latency -gt 50) -Or ($PINGCloudflare.latency -gt 50))
            ######################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G pass AND Cf pass) AND (G slow XOR Cf slow) ####################################################################
            ######################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "2" + [char]09 + "22  " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "full connectivity, partial latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert2}
                }
            Else
            ######################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G pass AND Cf pass) AND (G fast AND Cf fast) ####################################################################
            ######################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "0" + [char]09 + "    " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "full connectivity, no latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                }
            }
        ElseIf (($PINGGoogle.status -ne "Success") -And ($PINGCloudflare.status -ne "Success"))
        ##########################################################################################################################################################
        ######### (HR pass AND HR fast) AND (G fail AND Cf fail) ##################################################################################################
        ##########################################################################################################################################################
            {
            # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
            $statusLine = $nowLine + [char]09 + "4" + [char]09 + "4444" + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "no connectivity"
            $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
            $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
            If ($soundEnabled) {fun_alert4}
            }
        ElseIf (($PINGGoogle.status -ne "Success") -And ($PINGCloudflare.status -eq "Success"))
        ##########################################################################################################################################################
        ######### (HR pass AND HR fast) AND (G fail AND Cf pass) ##################################################################################################
        ##########################################################################################################################################################
            {
            If ($PINGCloudflare.latency -gt 50)
```

```
            ##########################################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G fail AND Cf pass) AND (Cf slow) ##################################################################################################
            ##########################################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "3" + [char]09 + "333 " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "partial connectivity, with latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert3}
                }
            Else
            ##########################################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G fail AND Cf pass) AND (Cf fast) ###################################################################################################
            ##########################################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "2" + [char]09 + "22  " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "partial connectivity, with no latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert2}
                }
            }
        ElseIf (($PINGGoogle.status -eq "Success") -And ($PINGCloudflare.status -ne "Success"))
        ##########################################################################################################################################################################
        ########## (HR pass AND HR fast) AND (G pass AND Cf fail) ##################################################################################################################
        ##########################################################################################################################################################################
            {
            If ($PINGGoogle.latency -gt 50)
            ##########################################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G pass AND Cf fail) AND (G slow) ######################################################################################################
            ##########################################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "3" + [char]09 + "333 " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "partial connectivity, with latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert3}
                }
            Else
            ##########################################################################################################################################################################
            ########## (HR pass AND HR fast) AND (G pass AND Cf fail) AND (G fast) ######################################################################################################
            ##########################################################################################################################################################################
                {
                # DateTime Severity Icon HR: Latency Status G: Latency Status Cf: Latency Status Summary
                $statusLine = $nowLine + [char]09 + "2" + [char]09 + "22  " + [char]09 + [char]09 + "HR:" + [char]09 + $PINGHomeRouter.latency + [char]09 + $PINGHomeRouter.Status +
[char]09 + [char]09 + "G:" + [char]09 + $PINGGoogle.latency + [char]09 + $PINGGoogle.Status + [char]09 + [char]09 + "Cf:" + [char]09 + $PINGCloudflare.latency + [char]09 +
$PINGCloudflare.Status + [char]09 + [char]09 + "partial connectivity, with no latency"
                $statusLine | Out-File -FilePath $fullOutFilePathAndName -Encoding ASCII -Append
                $statusLine | Out-File -FilePath $ISPProofOutFilePathAndName -Encoding ASCII -Append
                If ($soundEnabled) {fun_alert2}
                }
            }
        }
        $progressInDays = ([Math]::Round((($iterationCounter * $secondsBetweenTests) / $secondsPerDay),1))
        Write-Host $nowLine ", Interval:" $secondsBetweenTests "seconds , Iteration:" $iterationCounter "/" $maxIterations ", Days:" $progressInDays "/" $maxDaysToRun
        Start-Sleep -s $secondsBetweenTests
    } While ($iterationCounter -lt $maxIterations)
##############################################################################################################################################################################
```

Importing the Output TSV into Excel to Graph the Results:

(jump to TOC)

You can simply import the output TSV file into Excel, then select columns A and B, and insert a scatter chart:



Date and Time	Severity	Home Rou	HR Latenc	HR Status	Google	G Latency	G Status	Cloudflare	Cf Latency	Cf Status	Summary
5/20/2021 16:25	0	HR:	1	Success	G:	23	Success	Cf:	24	Success	full connectivity, no latenc
5/20/2021 16:25	0	HR:	2	Success	G:	25	Success	Cf:	24	Success	full connectivity, no latenc
5/20/2021 16:26	0										
5/20/2021 16:26	0										
5/20/2021 16:26	0										
5/20/2021 16:26	0										
5/20/2021 16:27	0										
5/20/2021 16:27	0										
5/20/2021 16:27	0										
5/20/2021 16:27	0										
5/20/2021 16:28	0										
5/20/2021 16:28	0										
5/20/2021 16:28	0										
5/20/2021 16:28	0										
5/20/2021 16:29	0										
5/20/2021 16:29	0										
5/20/2021 16:29	2										
5/20/2021 16:29	0										
5/20/2021 16:30	0										
5/20/2021 16:30	0										
5/20/2021 16:30	0										
5/20/2021 16:30	0										
5/20/2021 16:31	0										
5/20/2021 16:31	0										