What was my IP? Ask DoSvc on Windows 10

Gabriele Zambelli

Published on: Mar 31, 2019
DOI: 10.21428/3296d349
License: Creative Commons Attribution 4.0 International License (CC-BY 4.0)
Synopsis

**Forensic question:** What was the public IP address of a Windows 10 device?

**OS Version:** Windows 10

**File:** Event Trace Logs (ETL)

**Tools:** Powershell

Introduction

I recently watched the recording of the interesting talk *Windows Forensics: Event Trace Logs* that Nicole Ibrahim gave at SANS DFIR Summit 2018. I then used the tool **ETLParser** to dump the contents of all the ETL files stored on my own workstation. I glanced through the giant CSV output file looking for anything of interest and accidentally noticed a string "ExternalIpAddress". As you can see, the string is followed by an IP address.

That IP address is my current public IP address. Why is it there?
The string "ExternalIpAddress" is located next to other interesting strings like "GEO: response", "CountryCode" and a precious timestamp. **If this is a geolocation response, what triggered it?** Since "ExternalIpAddress" appears several times in the log files, how many geolocation requests have been made so far and why?

**DoSvc - Delivery Optimization**

All the hits were found within some logs whose names begin with "dosvc". I searched on Google and found out that "dosvc" stands for Delivery Optimization which is the update delivery service for Windows 10 clients. In the online documentation [Optimize Windows 10 update delivery](https://docs.microsoft.com/en-us/windows/client-management/deploy/deploy-over-the-network), Microsoft explains what this service does:

> Delivery Optimization is a new peer-to-peer distribution method in Windows 10. Windows 10 clients can source content from other devices on their local network that have already downloaded the updates or from peers over the internet.

Depending on the version of Windows 10, the various Event Trace Log (ETL) files created by the Delivery Optimization service (DoSvc) are stored here:

<table>
<thead>
<tr>
<th>OS Version</th>
<th>Default path</th>
<th>Filename</th>
</tr>
</thead>
<tbody>
<tr>
<td>Win10 (1507)</td>
<td>C:\Windows\Logs\dosvc</td>
<td>dosvc.\d*.\d.etl</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(e.g. dosvc.13777651.etl)</td>
</tr>
</tbody>
</table>
On my computer running Win10 (1803), which is always on and connected to the internet 24/7, the default DoSvc log path contains each day about 140 log files. Based on what I've observed, Win10 (1803) daily removes the ETL files older than 57/58 days. In such a scenario, there may be the chance of extracting several public/external IP addresses from the logs.

Even though the log files lead in the direction of "Delivery Optimization", I think something else might be responsible for the geolocation calls. On my computer, the "Delivery Optimization" service is off.
Delivery Optimization

Windows Update Delivery Optimization provides you with Windows and Store app updates and other Microsoft products quickly and reliably.

Allow downloads from other PCs

If you have an unreliable Internet connection or are updating multiple devices, allowing downloads from other PCs can help speed up the process.

If you turn this on, your PC may send parts of previously downloaded Windows updates and apps to PCs on your local network or on the Internet. Your PC won’t upload content to other PCs on the Internet when you’re on a metered network.

Advanced options

By default, we’re dynamically optimizing the amount of bandwidth your device uses to both download and upload Windows and app updates, and other Microsoft products. But you can set a specific limit if you’re worried about data usage.

Download settings

- Limit how much bandwidth is used for downloading updates in the background
- Limit how much bandwidth is used for downloading updates in the foreground

Upload settings

- Limit how much bandwidth is used for uploading updates to other PCs on the Internet
- Monthly upload limit

Note: when this limit is reached, your device will stop uploading to other PCs on the Internet.

Even the "Location" service is turned off.
Method

How to parse the logs

After some trial and error trying to figure out what was the best way to extract the data I needed from the CSV output file created by ETLParser, I found out that Windows 10 has a built-in Powershell cmdlet named "Get-DeliveryOptimizationLog":

- Announcing Windows 10 Insider Preview Build 17074 for PC:
  

This cmdlet retrieves decoded logs for Delivery Optimization. If no parameter is given, the cmdlet will parse the default DoSvc path. The parameter "-Path" is required to parse other locations:
Get-DeliveryOptimizationLog -Path C:\CustomPath\*  

Here I used the cmdlet to search for the keyword "ExternalIpAddress".

```
PS> Get-DeliveryOptimizationLog | Where-Object Message
-Like "*ExternalIpAddress*"
```

The output was:

```
TimeCreated : 27/11/2018 11:05:47
ProcessId : 13104
ThreadId : 9952
Level : 4
LevelName : Info
Function : CGeoInfoProvider::RefreshConfigs
LineNumber : 58
```

27/Nov/2018 11:05:47

I also spotted the IP that I was assigned to when using CyberGhost VPN.

```
TimeCreated : 24/11/2018 23:26:39
ProcessId : 36192
ThreadId : 3560
Level : 4
LevelName : Info
Function : CGeoInfoProvider::RefreshConfigs
LineNumber : 58
```

24/Nov/2018 23:26:39

What triggered the geolocation requests? Using the two examples shown above, I searched for "ProcessId" 13104 and 36192 and noticed that some events contain the message: "Create job name = WU Client Download".

```
TimeCreated : 27/11/2018 11:05:47
ProcessId : 13104
ThreadId : 9952
Level : 4
LevelName : Info
Message : Create job name = WU Client Download, jobId = 4d66d186-68e2-4bfc-8d74-f40de415fc
```

27/Nov/2018 11:05:47
What was downloaded and probably installed by the "Windows Update" (WU) client?

```
PS> Get-WuaHistory | Format-Table

This seems to be the answer:

<table>
<thead>
<tr>
<th>Result Date</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Succeeded</td>
<td>Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.281.899.0)</td>
</tr>
<tr>
<td>Succeeded</td>
<td>Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.281.899.0)</td>
</tr>
<tr>
<td>Succeeded</td>
<td>Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.281.756.0)</td>
</tr>
<tr>
<td>Succeeded</td>
<td>Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.281.756.0)</td>
</tr>
</tbody>
</table>

I see from my Windows Update history that Windows Defender Antivirus is updated on a daily basis. Based on the log files, it seems that "WU Client" makes a geolocation call before downloading any available update. That could explain why I have at least one geolocation response per day in the logs. Additionally, the creation time (TimeCreated) of each "GEO: response" event message always matches or is very close to the installation date of each antivirus definition update.

A hive file named "dosvcState.dat" is also involved in the process, but I haven't had the time yet to check what it contains. The hive can be found here:

C:\WINDOWS\ServiceProfiles\NetworkService\AppData\Local\Microsoft\Windows\DeliveryOptimization\State\dosvcState.dat
Scripting

I wrote a Powershell script that adapts to my needs the output provided by the mentioned cmdlet. The script adds to the output the name of the log files from which the information was extracted and shows the contents of the "Message" object in a different way. The script will generate two output files in both CSV and JSON format:

- `<timestamp>_dosvc_ExIpAddress`: contains the extraction of each IP found in the ETL files;
- `<timestamp>_dosvc_ip2location`: contains additional details about each unique IP found like the Internet service provider, latitude and longitude. The script relies on the public API provided by ip-api. At the time of this writing, the API is free for non-commercial use and needs no API key. The syntax to run a query is: `http://ip-api.com/json/{ipaddress}`.

Filenames are prepended with the timestamp of when the script was executed. To use the script, just provide the path containing the ETL files to parse:

```
PS> .\Get-DosvcExternalIP.ps1 C:\LogPath
```

From the logs of the computer I mentioned above (Win10 - 1803), I managed to extract 124 IP addresses whose dates range from October 10th 2018 to December 5th 2018.

I also used the script against the DoSvc ETL files that I extracted from a laptop (Win10 - 1709) that I last used in May in Las Vegas during the Magnet User Summit 2018. I connected to the Wi-Fi network in the hotel a couple of times for a short moment, but that was long enough for Windows 10.

This is an example of what I could extract from the logs:

```
PS> (Get-Content 20181205_143620_dosvc_ExtIpAddress.json | ConvertFrom-Json) | Where-Object ExternalIpAddress -eq "xx.xxx.xx.x6"
```

```
LogName : C:\TEMP\ETL_Laptop2\dosvc.20180522_170803_773.etl
TimeCreated : 22/05/2018 17:09:04
ExternalIpAddress : xx.xxx.xx.x6
CountryCode : US
ProcessId : 7840
ThreadId : 776
Level : 4
LevelName : Info
KeyValue_EndpointFullUri : https://kv801-prod.do.dsp.mp.microsoft.com/all
Version : <omissis>
Function : CGeoInfoProvider::RefreshConfigs
LineNumber : 58
```
This is an example of the additional details provided by the script using the ip-api API:

```
PS> Get-Content 20181205_143620_dosvc_ip2location.json | ConvertFrom-Json
```

```
as : ASxxxxx Cox Communications Inc.
city : Las Vegas
country : United States
countryCode : US
isp : Cox Communications Inc
lat : xx.x892
lon : -xxx.x63
org : HOSPITALITY NETWORK, LLC
query : xx.xxx.xx.x6
region : NV
regionName : Nevada
status : success
timezone : America/Los_Angeles
zip : 89106
```

The script `Get-DoSvcExternalIP.ps1` can be downloaded from my GitHub repository:

https://github.com/forensenellanebbia/powershell-scripts

**Link to original blog post:**

https://forensenellanebbia.blogspot.it/2018/12/what-was-my-ip-ask-dosvc-on-windows-10.html

**DFIR Review**

This work informs readers about an alternative method to the traditional (ipconfig command on the command prompt) finding the public IP addresses of a computer during a particular time period. This work details how to extract and interpret the prior, public IP addresses assigned to a Windows 10 computer stored in the event trace logs (ETL) generated by the Delivery Optimization Service (DoSvc) associated with Windows Update.

The fact that the recorded IP address is the public one is of particular importance. Determining the public IP address of a device on a given day may be crucial for investigations into online crime cases where the computer of a suspect has been seized. The information described in this work could even
date back earlier than Internet Service Provider (ISP) records. In addition, the ETL entries provide an update to the networks to which the device was connected on a given date, providing information about the way the device was used to access the internet. This information could provide insights as to whether a computer was connected to a VPN with a certain IP address. For portable computers, this information could give insights about the possible locations the computer was used from. Testing indicates that the IP addressed is logged as part of a core functionality of Windows, in a periodical way, which enhances the usefulness of the finding.

The proposed use of the IP address to determine a location should be treated with caution. While it is true that geolocation of an IP address is generally possible, the precision of the information may vary wildly depending on the country and the provider. This imprecision became evident during my tests where at least one of the geolocations indicated the wrong town (approximately 70 km off).

Both of these uses of the public IP encounter a challenge when facing the possibility of mobile networking. This is most likely to be a problem when analyzing laptops, where the connection to the hotspot of a mobile device may be a recurring scenario. Mobile providers generally use PAT to increase the number of devices that can be used simultaneously on the same IP address. This situation leads to the same IP being used by thousands of devices at the same time. Therefore, this information is only of limited use when investigating the authors of online crime.

This work provides a Powershell script to extract information from the logs. Running this script on some systems retrieves the public IP address but does not access the public API. This error occurs because jq-win64.exe or jq-win32.exe (specific to the system 32bit or 64bit) must be installed in the system32 directory before Powershell can access the API.

**Future Work**

This work draws attention to potential future work extracting information from "dosvcState.dat" files. In addition to extracting information from the Dosvc ETL, possible corroborating details could be explored in network connection event logs such as Application and Service logs/Microsoft/Windows/NetworkProfile log.
**Reviewers**

- Addisu Afework Birhanu (Verified using Reviewer Generated Dataset)
- Timothy Bollé (Verified using Author Provided Dataset)
- Alex Ogbole (Verified using Reviewer Generated Dataset)
- Terrence Nemayire (Methodology Review)
- Francesco Servida (Verified using Reviewer Generated Dataset)
- Hannes Spichiger (Verified using Reviewer Generated Dataset)