Dial Plan Analyser

# Overview

Tool to analyse CUCM dial plan to find unused phone numbers (i.e. no DN, translation pattern, route pattern, etc. that matches it), requires Python 3.6+ to run.

Many Linux distros have Python installed by default. For Windows the easiest install is the official Python Windows version, or Miniconda works fine too:

Miniconda distribution of Python: <https://conda.io/miniconda.html>

Official Python distribution: <https://www.python.org/downloads/>

The lxml, Requests, urllib3 and Zeep libraries are required to work.

# Version History

(c) 2017 - 2019, Chris Perkins:

v1.5 – code tidying.

v1.4 – GUI adjustments & fixes some edge cases.

v1.3 – added AXL support.

v1.2 – bug fixes.

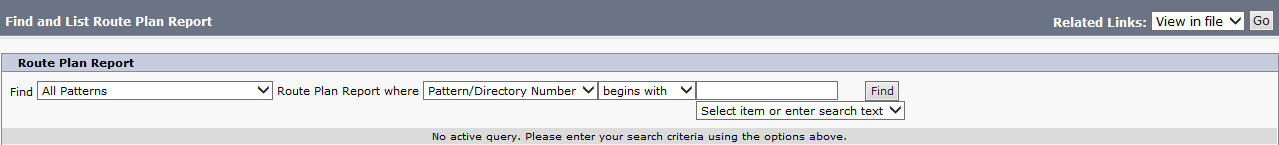
v1.1 – added GUI.

v1.0 – initial release with only CSV file support and CLI usage.

All testing was done using Windows. CSV files tested with CUCM v9.1 and v10.5, AXL tested with CUCM v11.5.

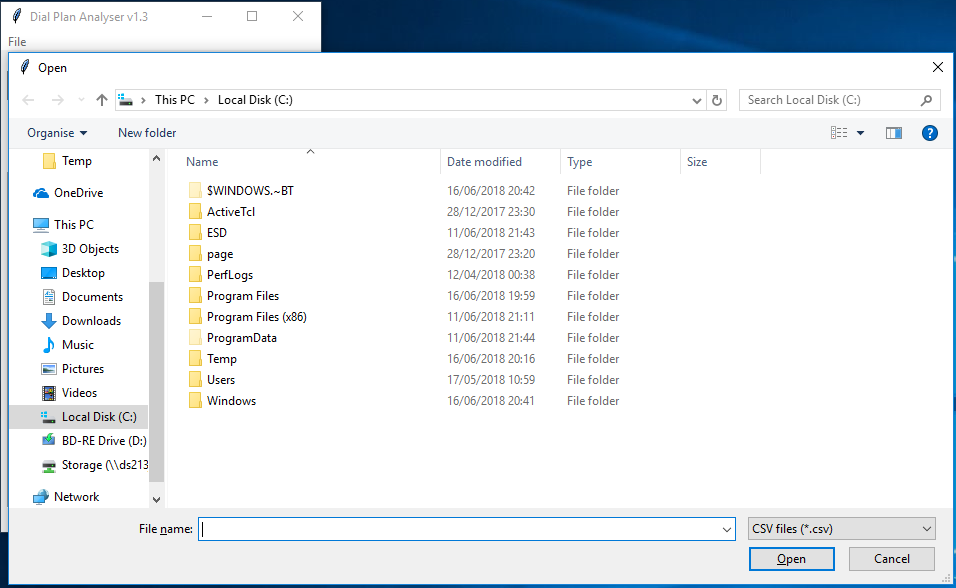
# Using With CSV Files

This method imports dial plan information from CUCM via CSV files. These are created from within CUCM via **Call Routing > Route Plan Report > View in file**:

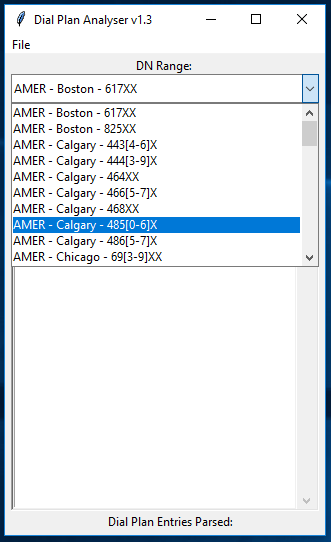


Therefore before using the tool, export the Route Plan Report from the CUCM cluster that you want to find unused numbers for.

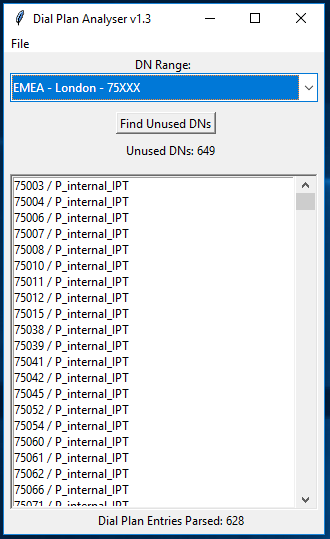
Load the CSV file via **File > Load CSV**:



Then select a direct dial range from the drop down list under **DN Range**:



Click **Find Unused DNs**, it will then process the CSV file and find numbers in the selected range that aren’t currently in use. The list of unused DNs is in the format *directory number* / *partition*, so you can easily see which numbers and which partition the search is working on:



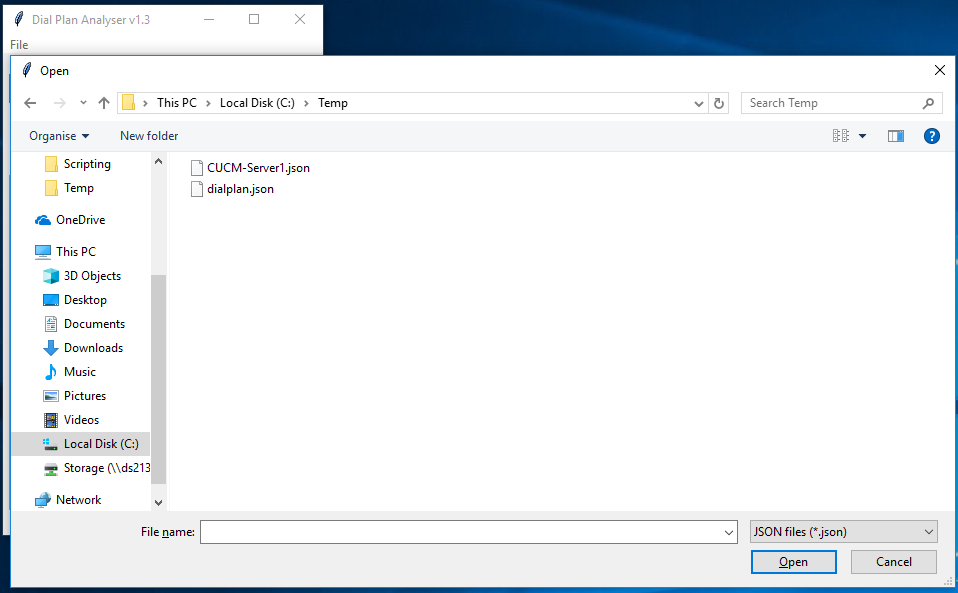
**Unused DNs** lists how many unused directory numbers were found during the dial plan analysis.

**Dial Plan Entries Parsed** lists how many possible numbers it had to analyse to find the unused DNs.

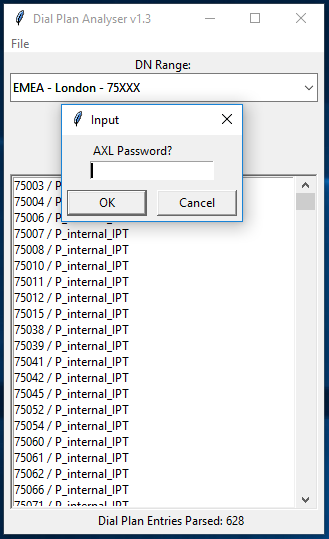
# Using with AXL

This method imports dial plan information from CUCM using the AXL API. The AXL schema for the version of CUCM in use is required, this is downloaded from CUCM via **Application > Plugins > Cisco AXL Toolkit**. The requires files contained within the .zip file are *AXLAPI.wsdl, AXLEnums.xsd* and *AXLSoap.xsd*.

Different CUCM servers are defined in JSON formatted files, allowing for multiple CUCM clusters running different versions (and thus different AXL schemas). Load the CSV file via **File > Load AXL**:



It will then prompt for the password:



After this the process is identical to working with CSV files.

# Customising the Tool

The direct dial ranges to search for can be customised, so that the tool can be used for any CUCM cluster. These settings are stored in *dialplan.json* in JSON format, for example:

[

{

"description": "ANZ - Sydney - 2XXXX",

"range\_start": "20000",

"range\_end": "29999",

"partition": "INTERNAL"

},

{

"description": "ANZ - Adelaide - 30[23]XX",

"range\_start": "30200",

"range\_end": "30399",

"partition": "INTERNAL"

}

]

* The JSON file starts with [ and ends with ].
* Each direct dial range is enclosed within { } and contains parameters for the description, range start, range end and partition. The field headings and values must be enclosed within “”.
* The range end must be greater than the range start.
* The direct dial ranges must have a comma after each, except for the last one.

So to add another range to the above example:

[

{

"description": "ANZ - Sydney - 2XXXX",

"range\_start": "20000",

"range\_end": "29999",

"partition": "INTERNAL"

},

{

"description": "ANZ - Adelaide - 30[23]XX",

"range\_start": "30200",

"range\_end": "30399",

"partition": "INTERNAL"

},

{

"description": "ANZ - Adelaide - 309XX",

"range\_start": "30900",

"range\_end": "30999",

"partition": "INTERNAL"

},

{

"description": "ANZ - Canberra - 33[1-3]XX",

"range\_start": "33100",

"range\_end": "33399",

"partition": "INTERNAL"

}

]

The parameters for using AXL are also stored in JSON format:

[

{

"fqdn": "cucm-emea-pub.somewhere.com",

"username": "AppAdmin",

"wsdl\_file": "file://C://temp//AXLAPI.wsdl"

}

]

* “fqdn” should be the FQDN or IP address of the target CUCM publisher.
* “username” is an application or end user with the Standard AXL API Access role.
* “wsdl\_file” points to the location of the AXL schema, note the slightly different path syntax for Windows.