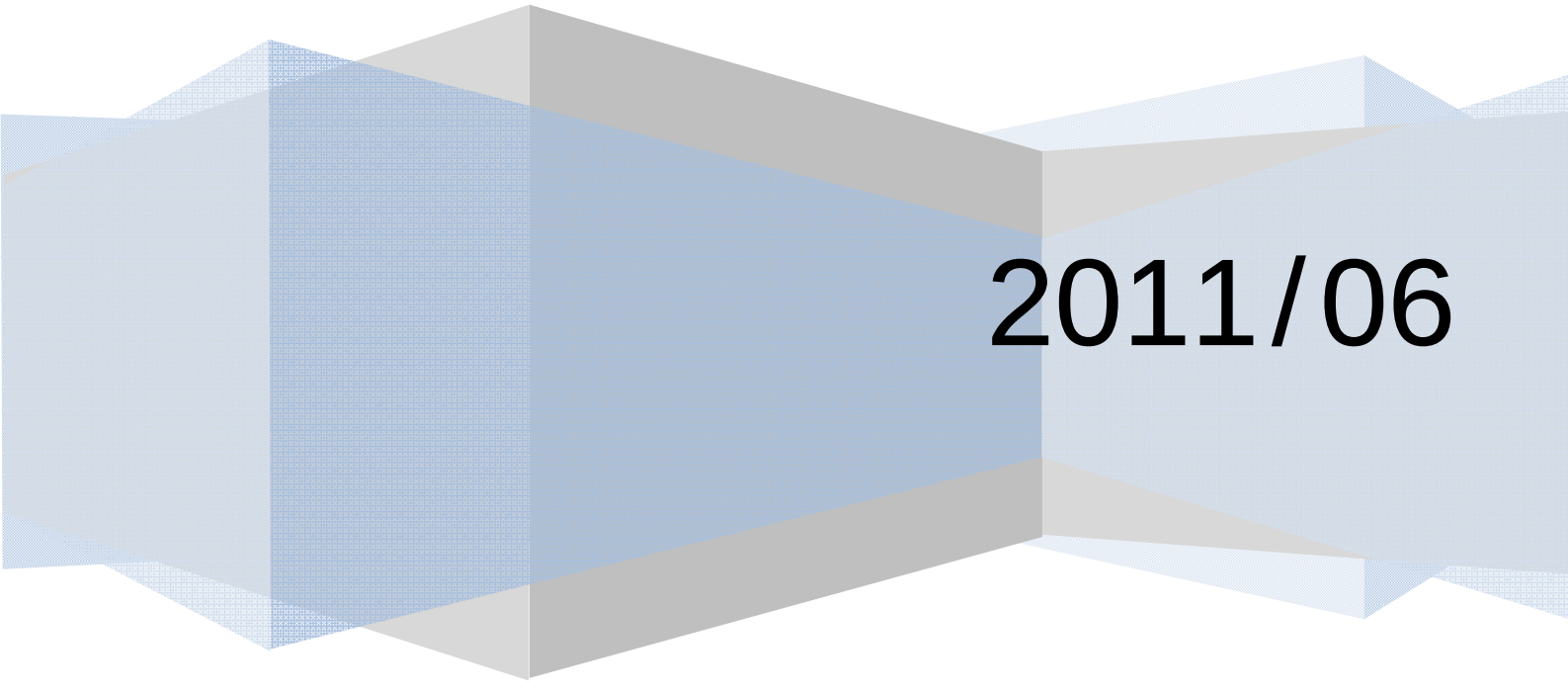
PATERVA

Maltego TDS Transforms

A reference guide

AM



2011/06

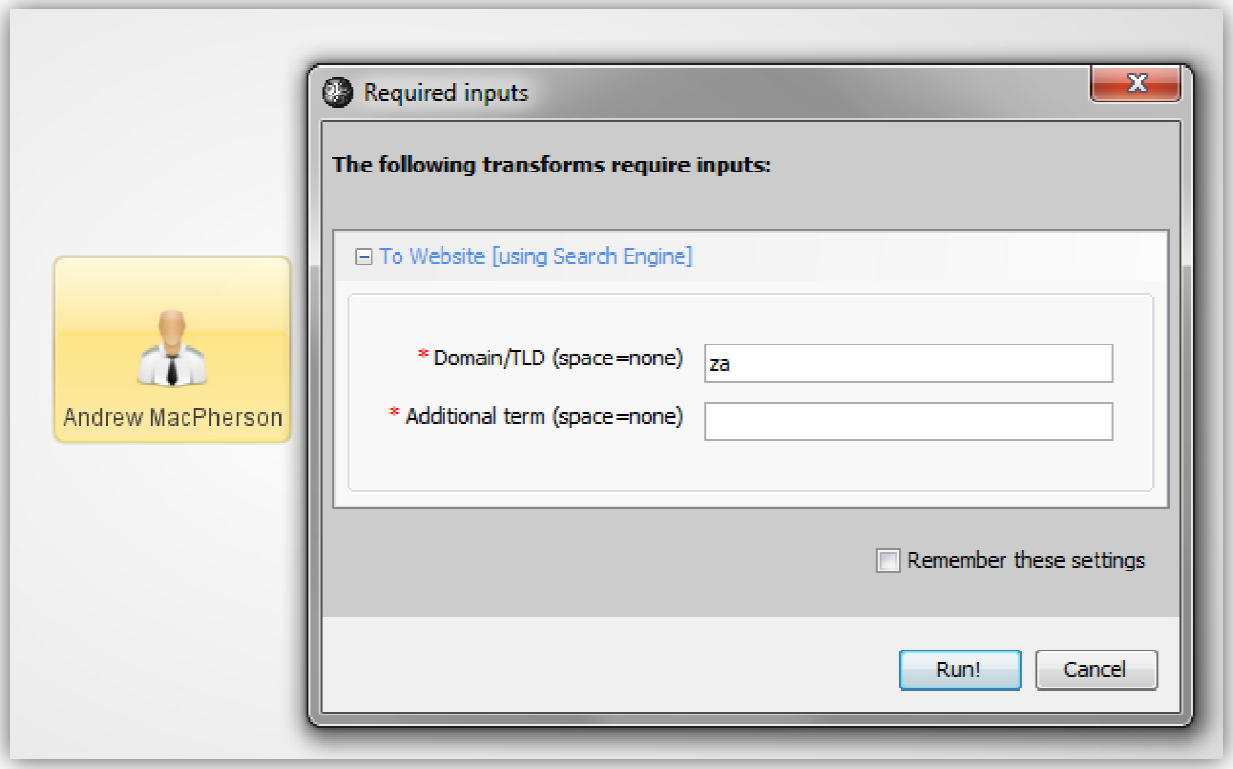
Table of Contents

1 Introduction

This document serves as a reference guide of transforms that are currently in use on the Transform Distribution Server (TDS) that can be found at http://cetas.paterva.com/TDS/. This guide only documents the transforms that Paterva has developed and does not include any that other individuals may have implemented.

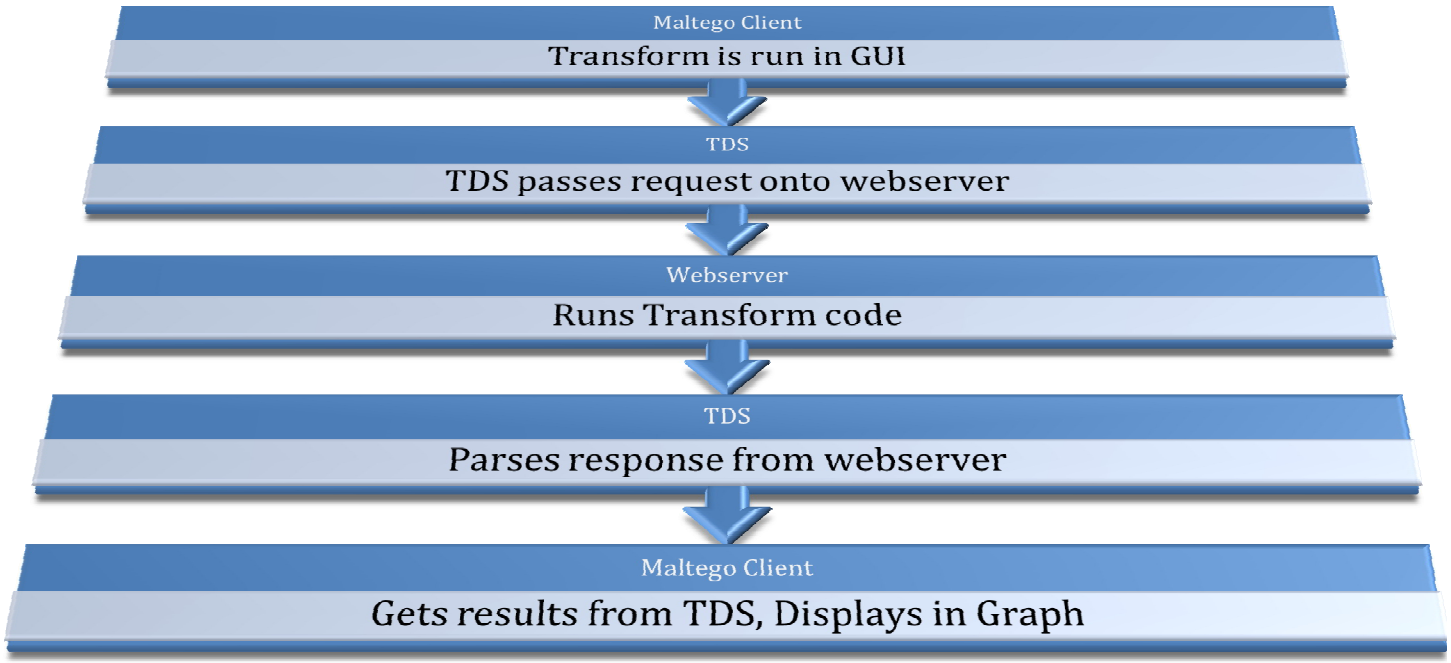
1.1 What is the TDS

The TDS or Transform distribution server came out of a need to rapidly develop and share transforms to a number of users without the hassle of using local transforms. TDS Transforms also provide the transform developer with the ability to gain additional information from the client via popup messages (seen below) and slider values:



1.2 TDS Infrastructure

The TDS can be likened to an HTTP proxy whereby the end client communicates through an application to the target Website. In the same way the TDS 'proxies' transform request to an end server and can be visualised as follows:



1.3 Benefits of the TDS

Having a TDS'd transform has many benefits including:

- No need to setup a development environment on every end users PC (you don't need python on every client)
- Code privacy/security – because code is now in a remote webserver you don't have to worry about end clients seeing things like database passwords.
- Easy Updating – code can be updated without any disturbance to end clients, i.e., make changes to outputted entities will automatically reflect on each client

1.4 Client Infrastructure

The Maltego client works in the same way with the TDS as it does with a normal server, via the discovery process, which are the follow steps:

1. Client is pointed to a Seed URL (contains Runner links)
2. Client then follows to Runner and identifies all Transforms on a Runner
3. Client can then execute transforms against the above Runner

You can think of Seed URLs as library guide on a bookshelf, telling you what all the books (Runners) on a specific shelf are. In this analogy the Runner would then be the book itself, containing an index telling you what chapters (Transforms) are in the book and how to get to them.

2 Transforms

The Transforms listed in the following sections are all of the transforms we have developed for public consumption up to date. These are divided into various Seeds based on transforms we feel should be grouped together. It should also be noted that various transforms require new entities to be fully used within Maltego.

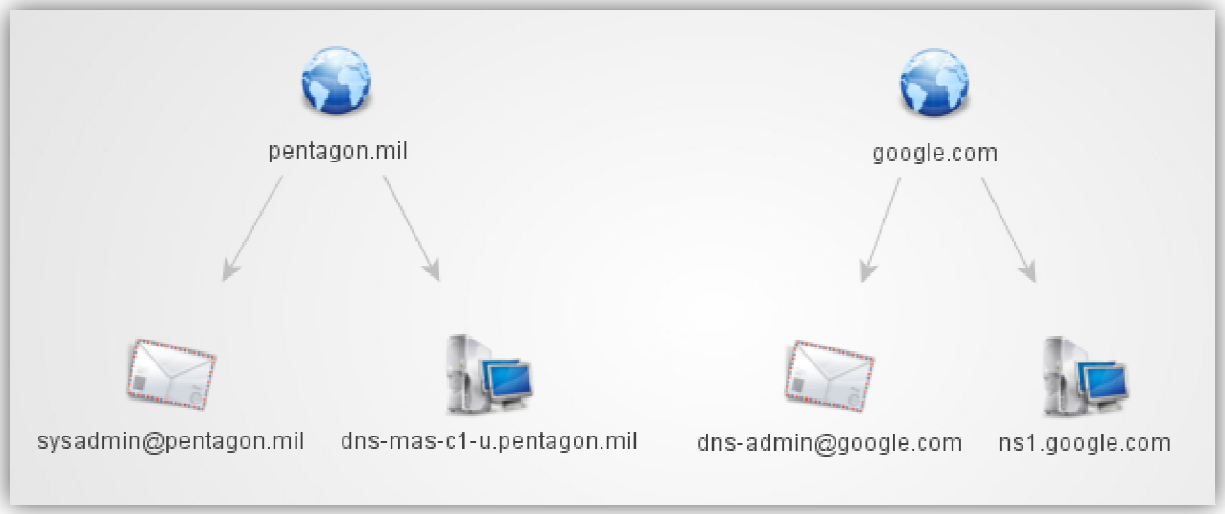
These transforms are listed by input type (the entity you right click on the graph) and have a full description as well as an example graph to show their usage.

2.1 Domain Entity

2.1.1 Domain to SOA Information

SOA (start of authority record) information on a domain gives you the ability to retrieve the primary name server and the email of the domain administrator for a specific domain. This is interesting to an analyst because often it can give you a machine (read DNS name) that you would not have previously found as well as an administrative Email Address.

See https://secure.wikimedia.org/wikipedia/en/wiki/List_of_DNS_record_types#SOA for more information on SOA records.

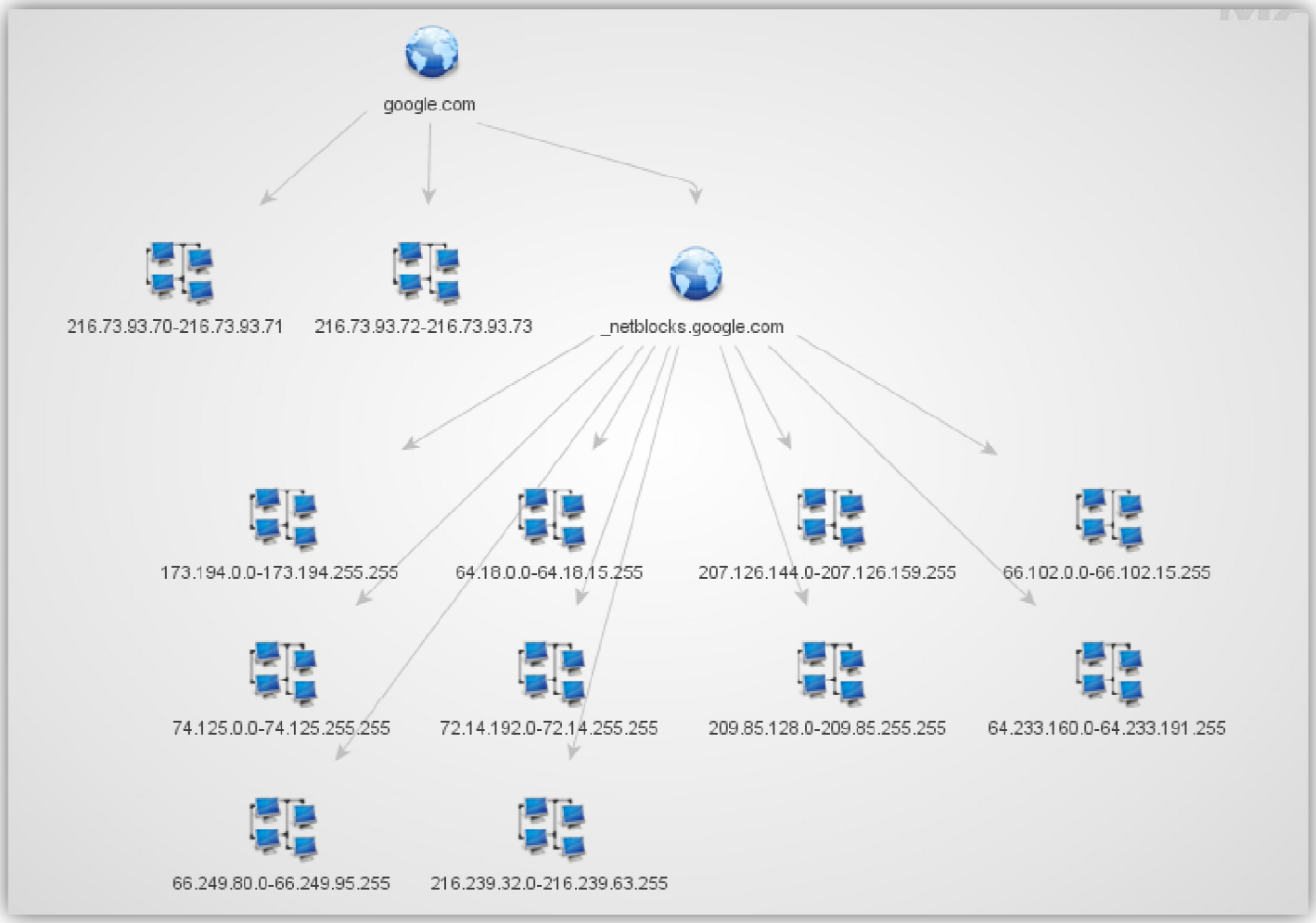


Additional Entities Required: None

2.1.2 Domain to SPF Information

SPF or sender policy framework was designed to help prevent spam by allowing DNS administrators to specify which IP addresses in their network are used to as mail servers for the domain. However SPF implementations are usually poorly done where entire network ranges rather than single machines are listed or even SPF specific subdomains are given. This is interesting to the analyst because one can enumerate a network range with a few transforms.

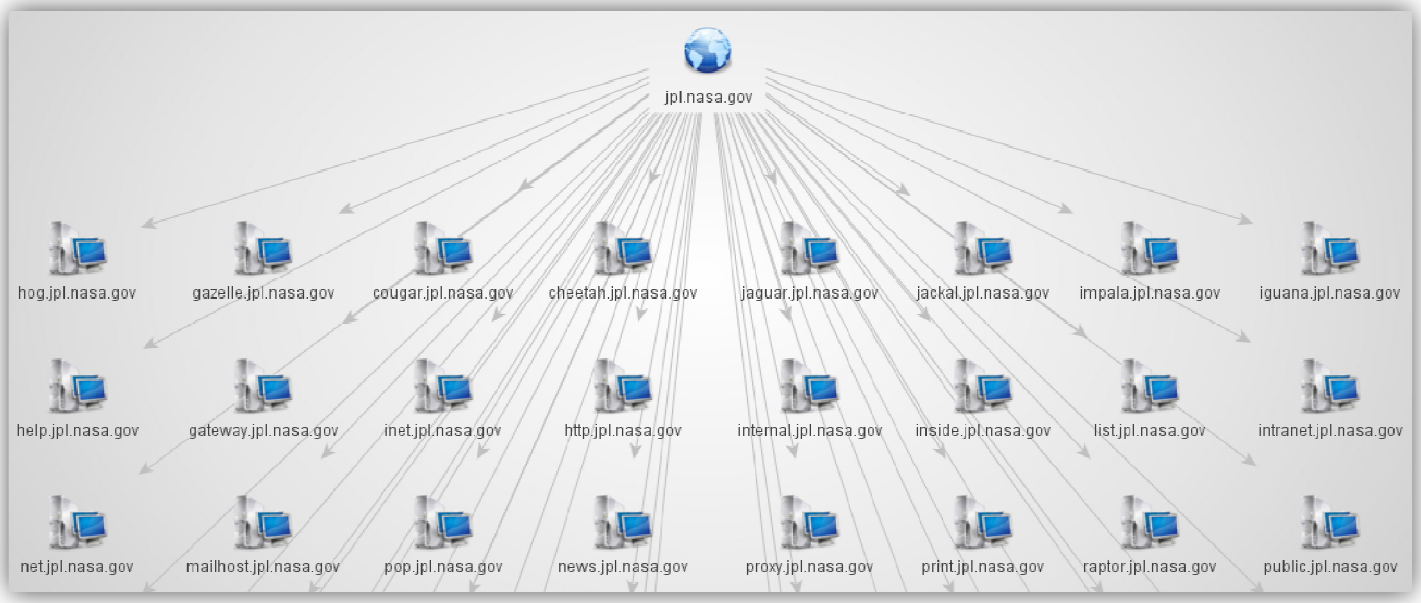
For more information on SPF records please see https://secure.wikimedia.org/wikipedia/en/wiki/Sender_Policy_Framework



Additional Entities Required: None

2.1.3 Domain to DNS Name Schema

This transform will try test various name schema's against a domain and try and identify a specific name schema for the domain. The default lists are stored at http://alpine.paterva.com/NSSchema.txt. As an example it will try a few names from a list of matrix characters and if the first few are found it will try the entire list, otherwise it will move onto the next list.



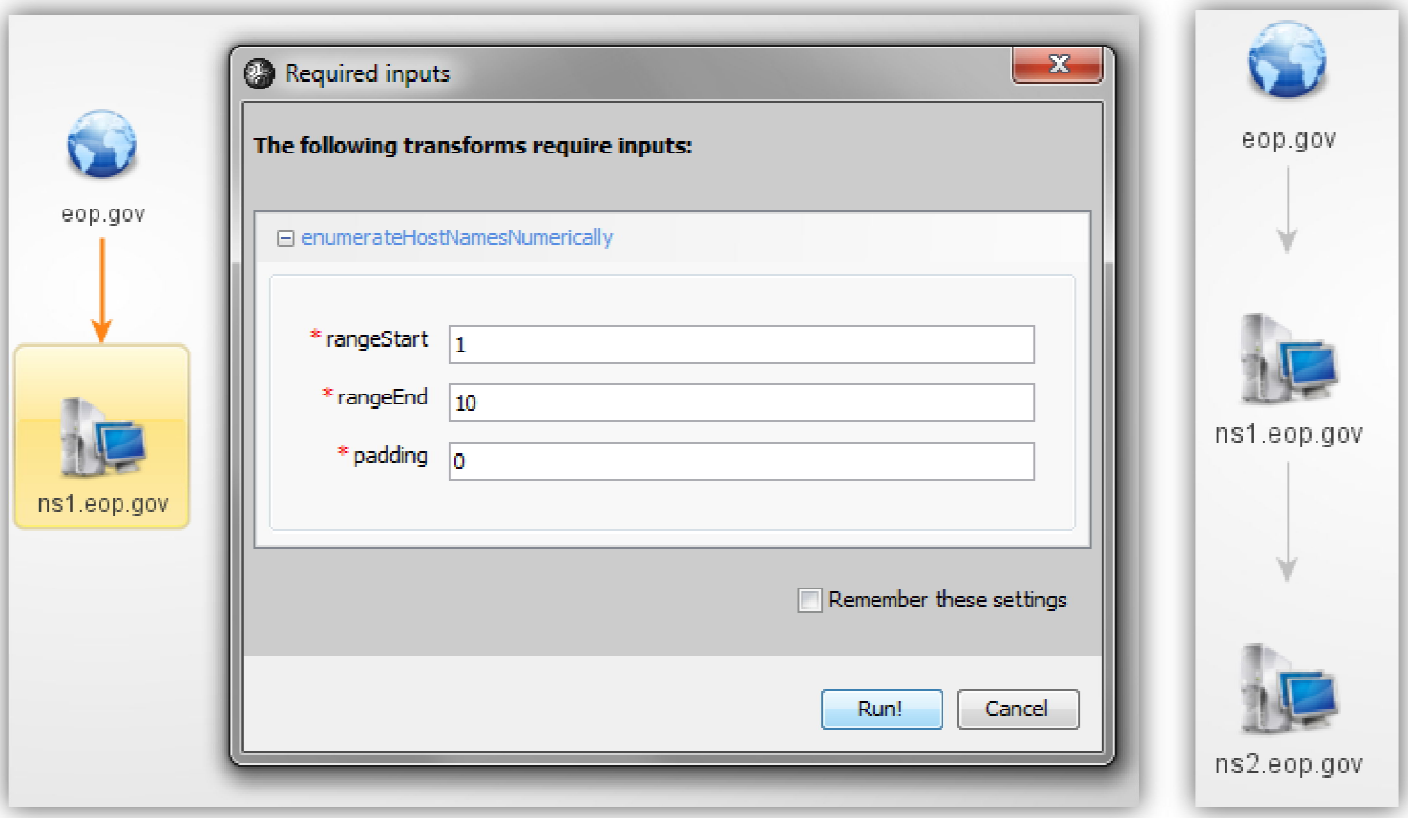
Additional Entities Required: None

2.2 DNSName

2.2.1 Enumerate host names numerically

This transform will test for the existence of DNS names that end with the same name, but another number. As example - if ran on mx1.domain.com it will check for mx1, mx2, mx3.domain.com. The range and padding can be set with transform settings.

This is useful to the analyst as one can enumerate many DNS names automatically.

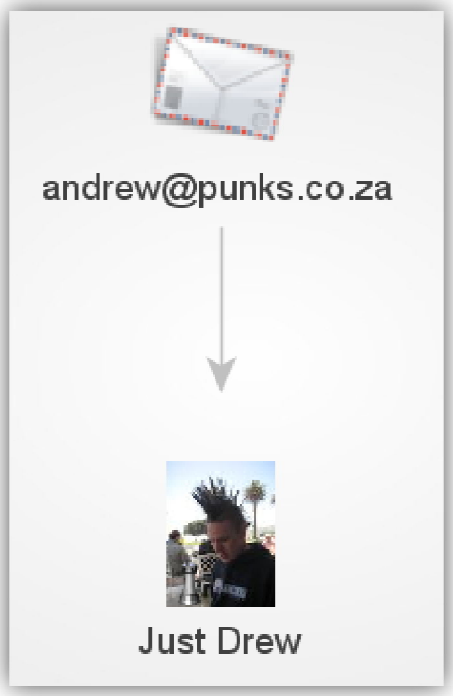


Additional Entities Required: None

2.3 Email Address

2.3.1 Email to MySpace Account

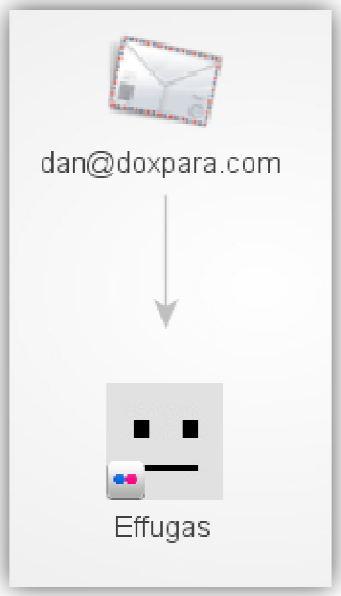
This transform merely resolves an email address to a MySpace affiliation. Useful when trying to enumerate social networks based on an individual's Email Address.



Additional Entities Required: None

2.3.2 Email to Flickr Account

This transform merely resolves an email address to a Flickr affiliation. Useful when trying to enumerate social networks based on an individual's Email Address.

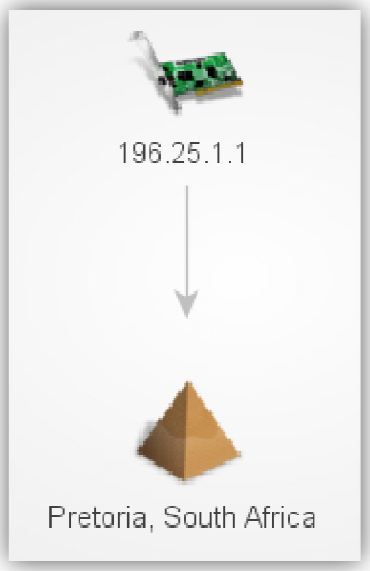


Additional Entities Required: None

2.4 IPv4 Address

2.4.1 To Location

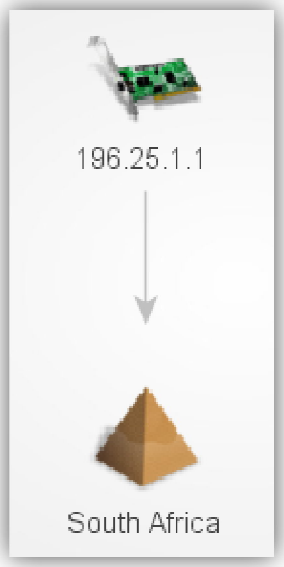
This transform uses MaxMind GeoIP Lite database to resolve an IP Address to a specific location. Useful for identifying in which locations specific IP address(es) are located.



Additional Entities Required: None

2.4.2 To Location Country

This transform does the same as the above transform although it only returns the country associated with the IP address. Useful for identifying in which countries specific IP address(es) are located.

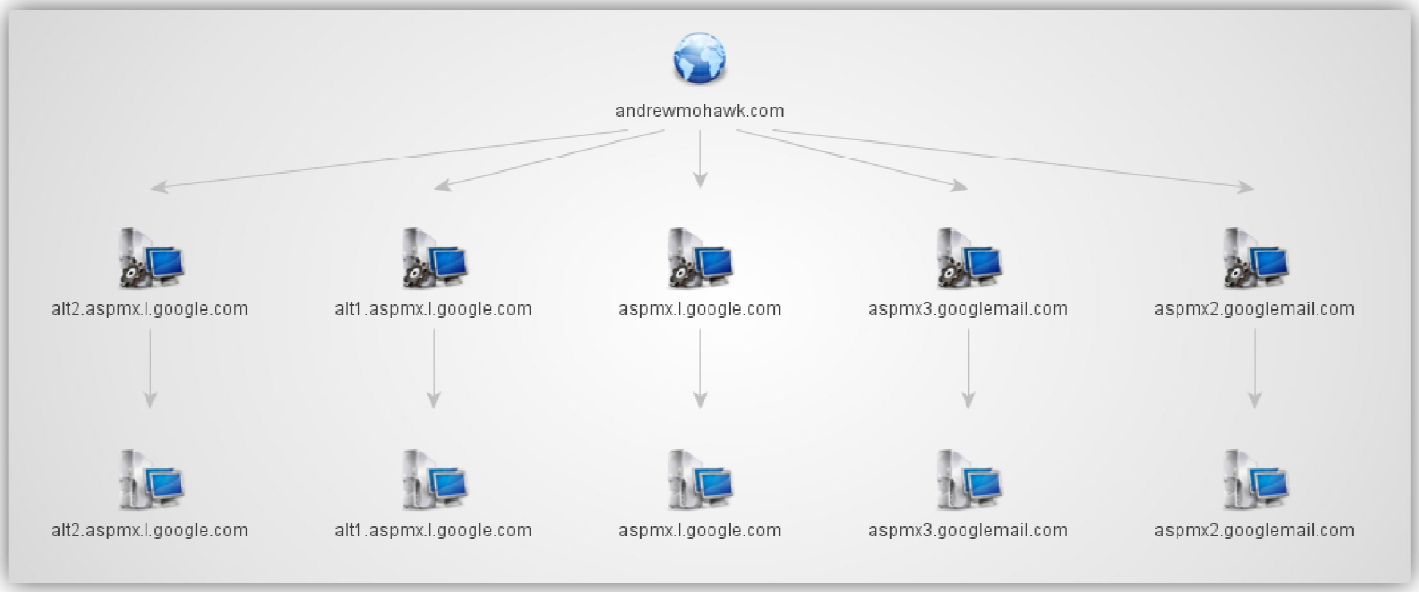


Additional Entities Required: None

2.5 MX Records

2.5.1 MX to DNS Name

This transform simply converts an MX record to a DNS record entity allowing you to use additional DNS transforms (such as numeric enumeration).

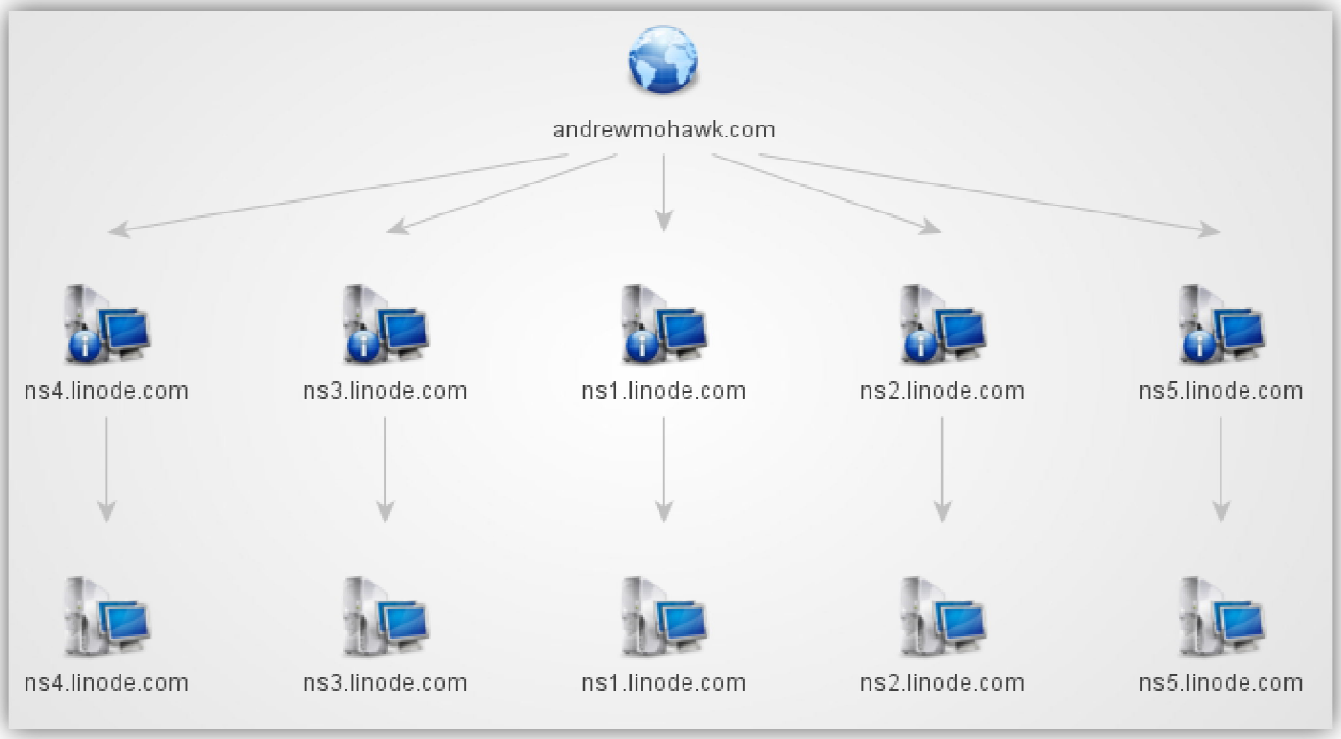


Additional Entities Required: None

2.6 NS Record

2.6.1 NS to DNS Name

This transform simply converts an NS record to a DNS record entity allowing you to use additional DNS transforms (such as numeric enumeration).

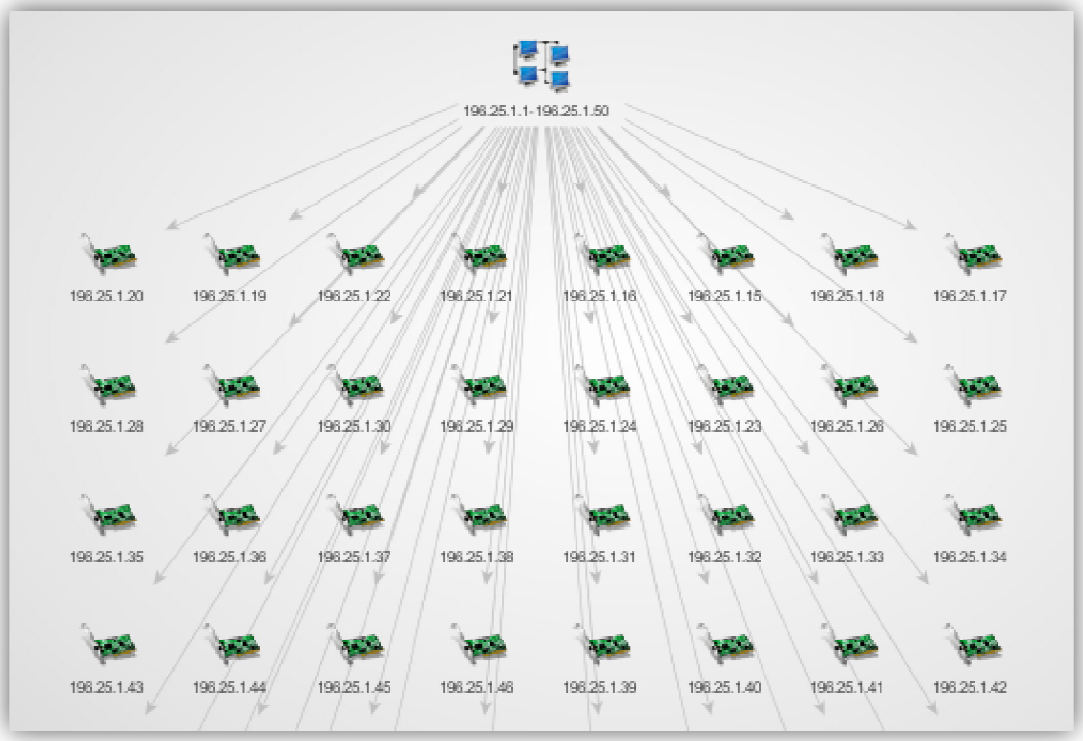


Additional Entities Required: None

2.7 Netblock

2.7.1 Netblock to IPs

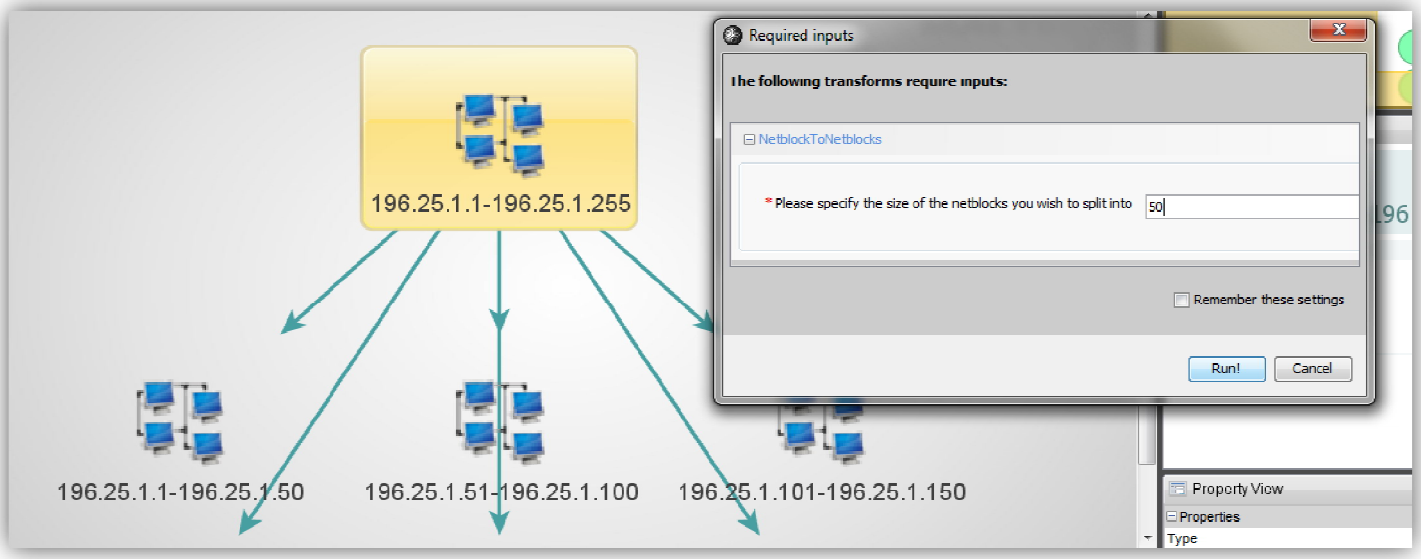
Simple transform to take a netblock to each of the individual IP addresses within it. Useful when doing things like trying to see if any of the IP addresses in a specific range where indexed by a search engine.



Additional Entities Required: None

2.7.2 Netblock to Netblocks

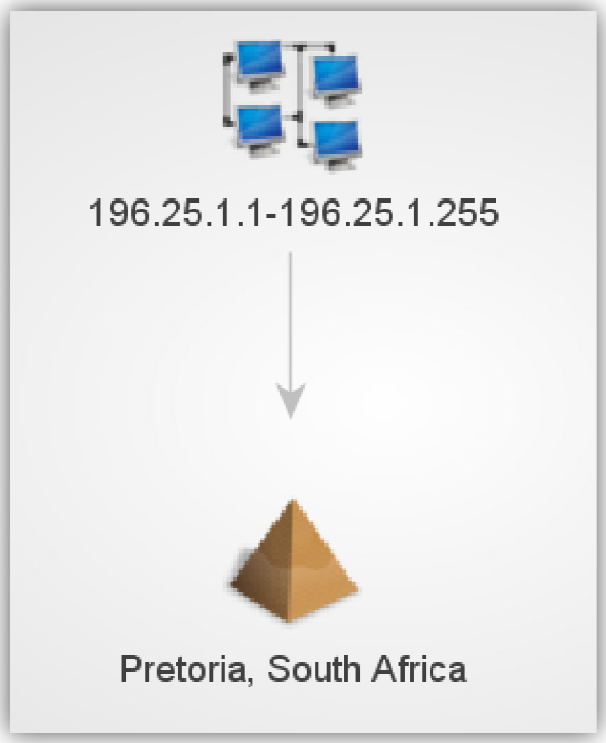
This transform splits a netblock into smaller netblocks so that they can be used with various additional transforms.



Additional Entities Required: None

2.7.3 To Location Netblock

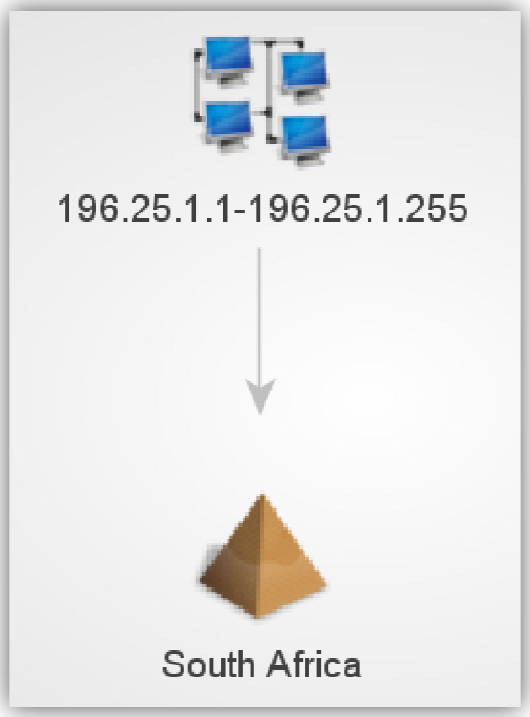
Using the MaxMind Geolite database this transform returns a location based on the netblock.



Additional Entities Required: None

2.7.4 To Location Netblock Country

Same as the previous transform using the MaxMind Geolite database to return just the country based on a netblock. Useful when trying to enumerate where a large portion of netblocks are from to narrow further analysis.

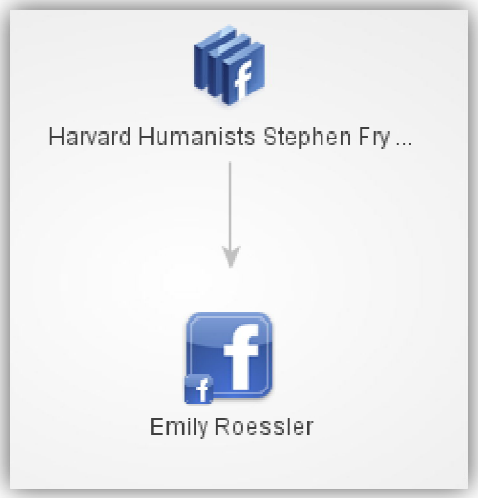


Additional Entities Required: None

2.8 Facebook Object

2.8.1 To Facebook Affiliation

This transform simply takes a Facebook graph object to a user profile of the user who made the post.



Additional Entities Required: FacebookObject

2.8.2 To Phrase

Facebook objects can be many things including links and other types and most of them contain a message, for example someone links to a website and writes something like “this is my favourite site evar”. This transform will return that message.

Additional Entities Required: FacebookObject

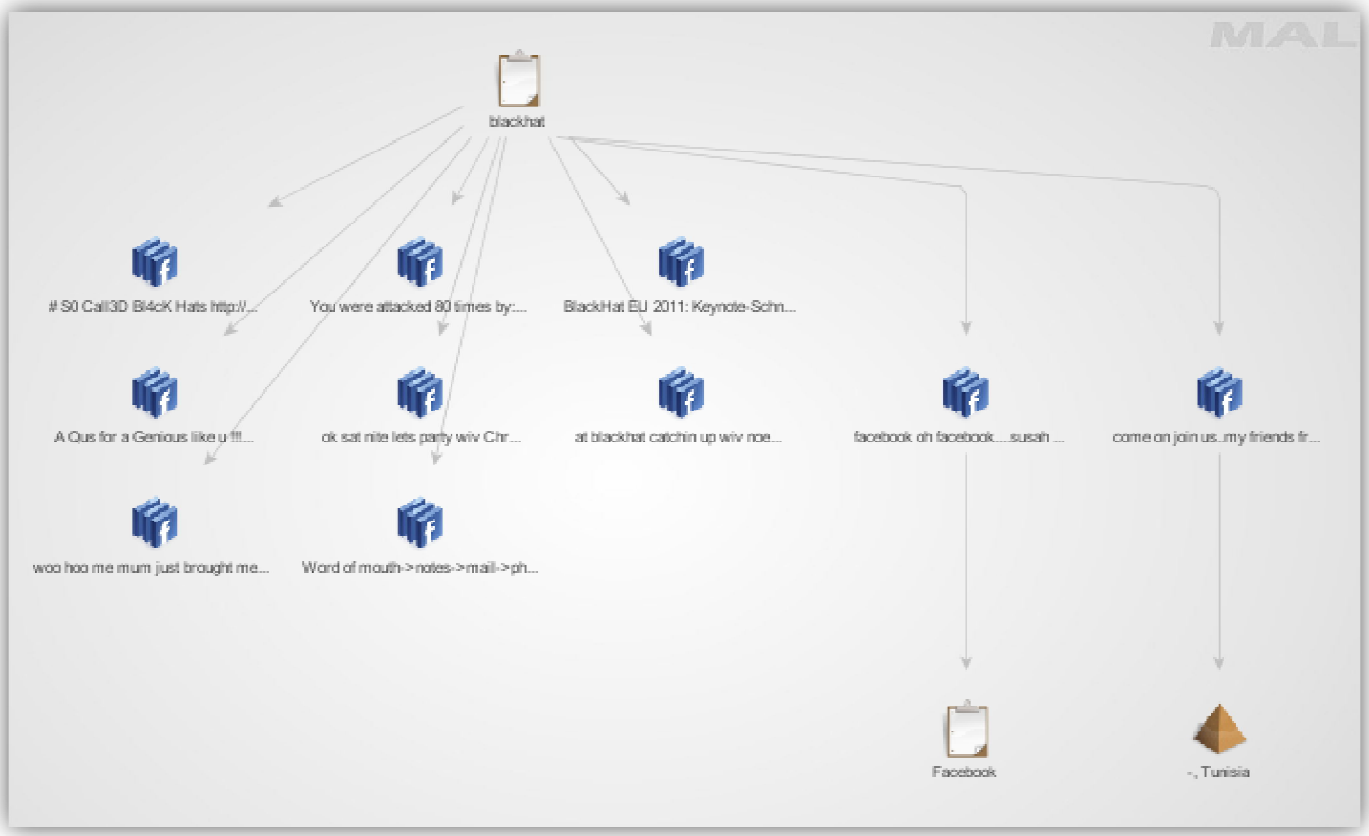
2.8.3 To Person from Facebook

This transform merely returns a person object of the person who made the post on Facebook.

Additional Entities Required: FacebookObject

2.8.4 To Entities NER

Using the 'message' found within a Facebook object, this transform will try identify various other entities by using Named Entity Recognition.

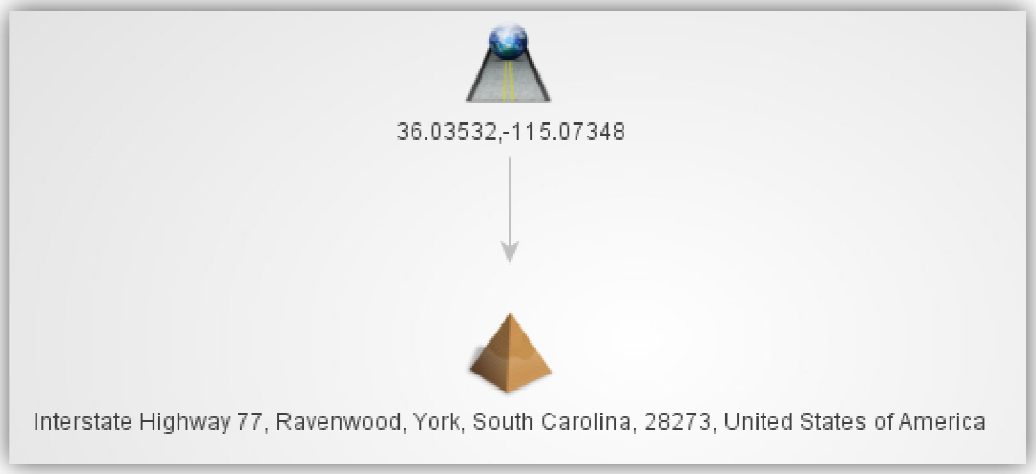


Additional Entities Required: FacebookObject

2.9 GPS

2.9.1 To Location GeoCode

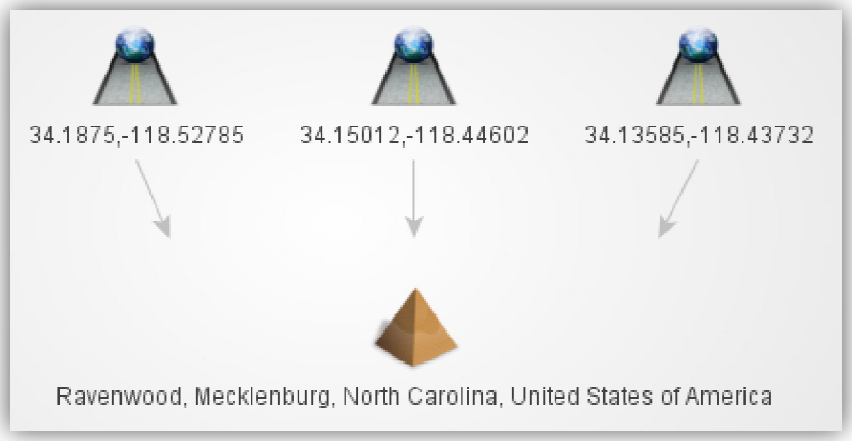
This transform uses the openstreetmap nominatim to take GPS co-ordinates to a physical location. This is useful for quickly identifying locations.



Additional Entities Required: GPS

2.9.2 To Location GeoCode Broad

This does the same as the above transform but returns a broader area, useful to identify GPS co-ordinates that are in a similar physical location.

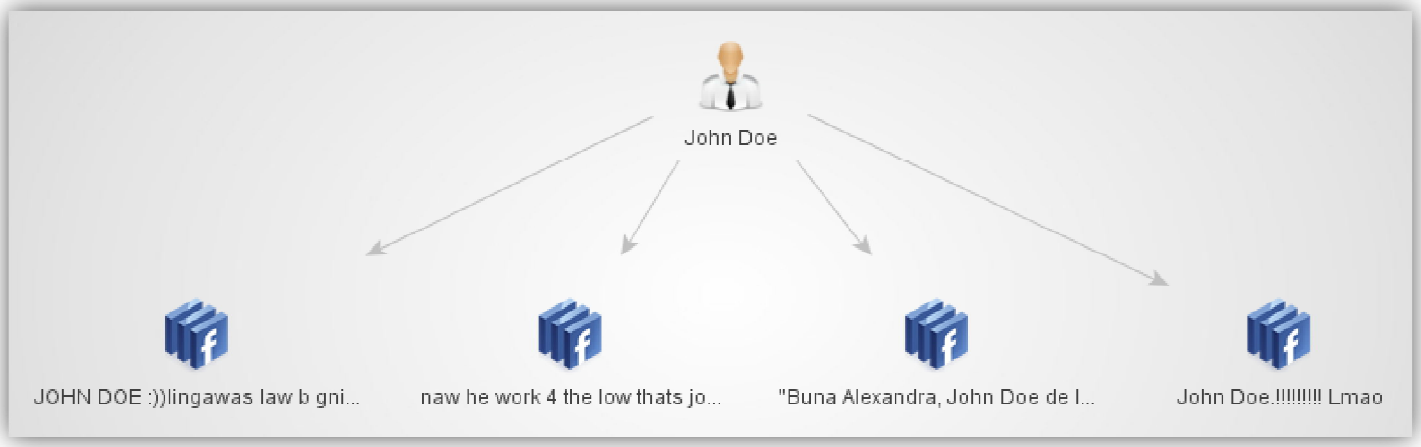


Additional Entities Required: GPS

2.10 Person

2.10.1 To Facebook Object Person

This transform searches Facebook's graphAPI for public posts relating to a person entity. Essentially it searches graphAPI for "firstname lastname".

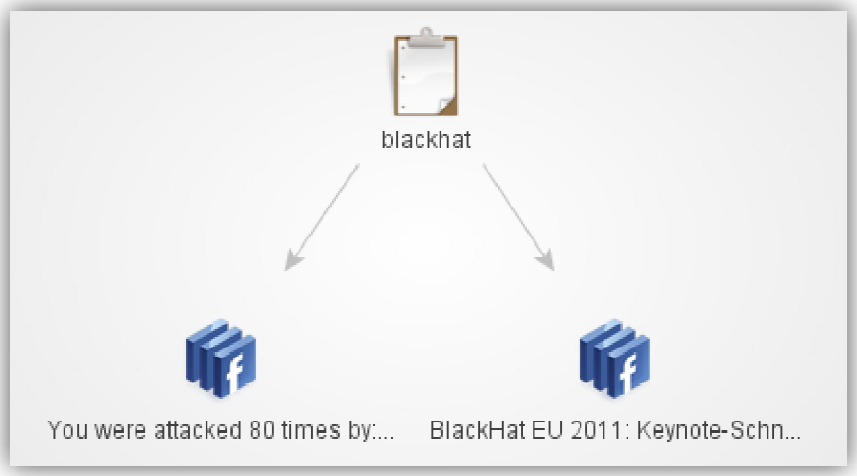


Additional Entities Required: FacebookObject

2.11 Phrase

2.11.1 To Facebook Object

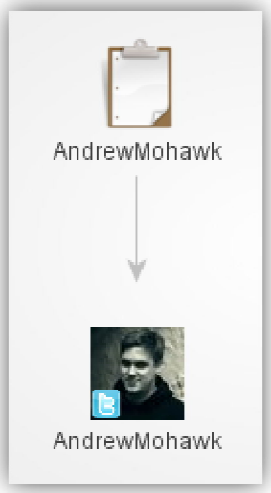
Searches Facebook's GraphAPI for the specific phrase, returning any posts found.



Additional Entities Required: FacebookObject

2.11.2 Alias to Twitter account

This transform is useful for getting a Twitter Affiliation for a known alias, looks up the alias on twitter and populates all the relative fields.



Additional Entities Required: None

2.11.3 To Facebook Profile

This transform is useful for getting a Facebook Affiliation for a known alias, looks up the alias on Facebook and populates all the relative fields.

Additional Entities Required: None

2.12 Twit

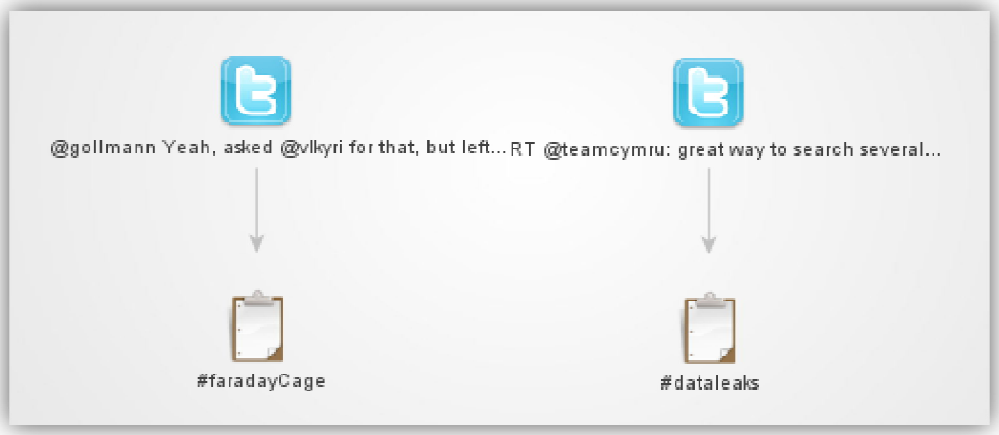
2.12.1 To Entities NER Twitter

This transform takes a tweet and identifies any additional entities used within it via Named Entity Recognition. Useful for finding similar concepts between tweets!

Additional Entities Required: None

2.12.2 Pull Hash Tags

Simple transform to pull hash tags from tweets, useful for identifying common hash tags across multiple tweets.



Additional Entities Required: None

2.12.3 Pull URLs

Same as the above transform apart from pulling URLs and expanding them where possible.

Additional Entities Required: None

2.12.4 Tweet Geo Info

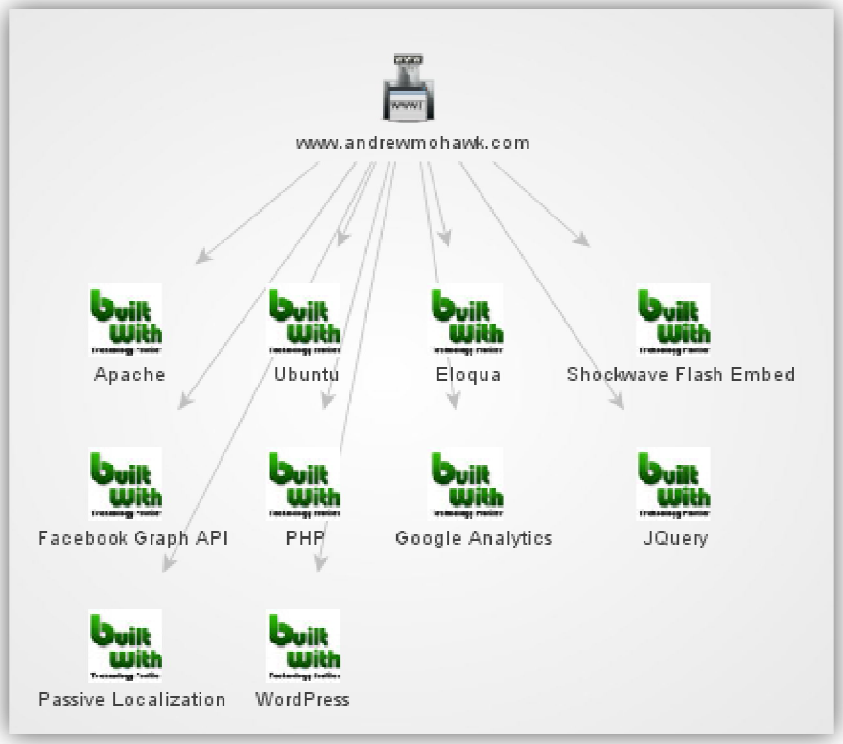
Returns any GPS co-ordinates found within specific tweets.

Additional Entities Required: GPS

2.13 URL

2.13.1 To Server Technologies URL

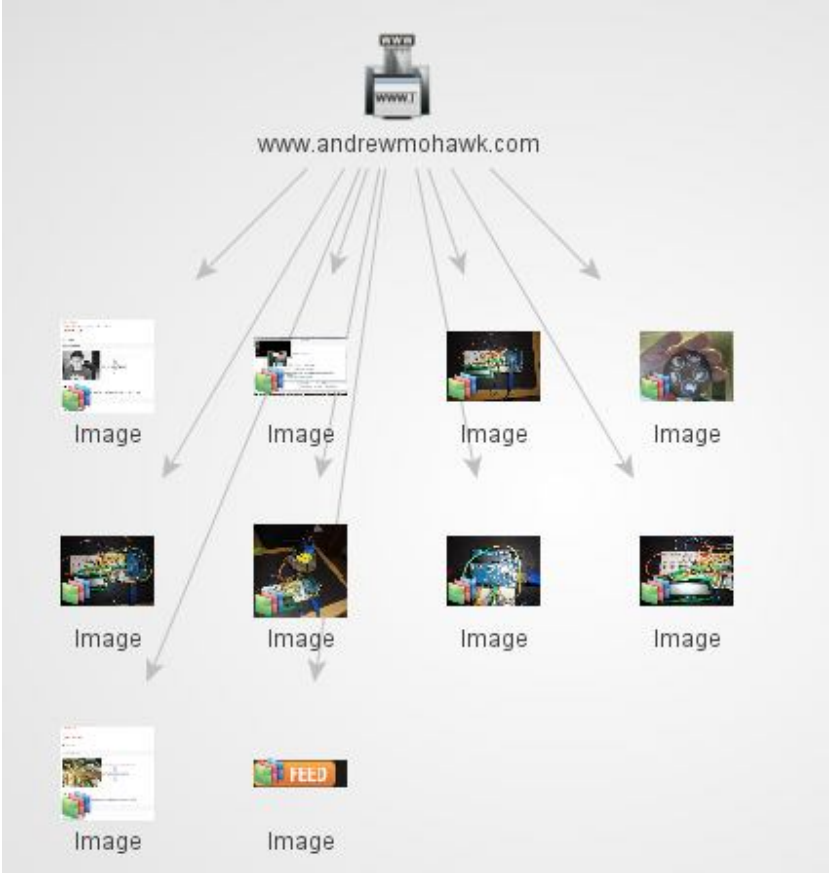
This transform utilises builtwith.com to identify common server technologies located on a specific URL. Useful when analysing multiple URLs to find anomalies within technologies used.



Additional Entities Required: BuiltWith

2.13.2 To Images from URL

This transform extracts all images from a specific website and displays them as image entities. This is useful for reverse image searching as well as looking for exif information within them. This transform also has a setting specifying the minimum image size. This is used to exclude pictures that are used within websites for things like bullets and icons.



Additional Entities Required: Image

2.14 Website

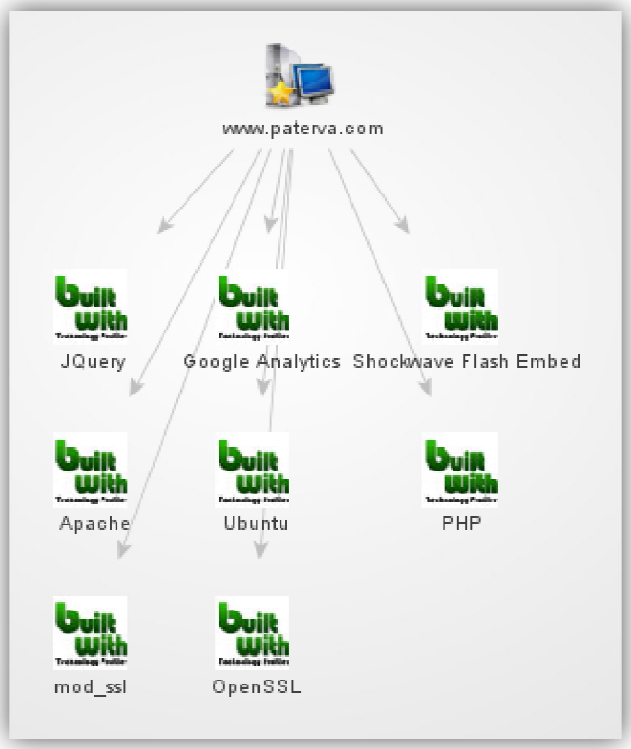
2.14.1 Website to DNS Name

Simply takes a website to a DNS name so that it can be used with other transforms such as enumerating hostnames numerically.

Additional Entities Required: None

2.14.2 To Server Technologies Website

Identifies server technologies in use on a specific Website. Useful when analysing multiple URLs to find anomalies within technologies used.



Additional Entities Required: BuiltWith

2.15 Affiliation Facebook

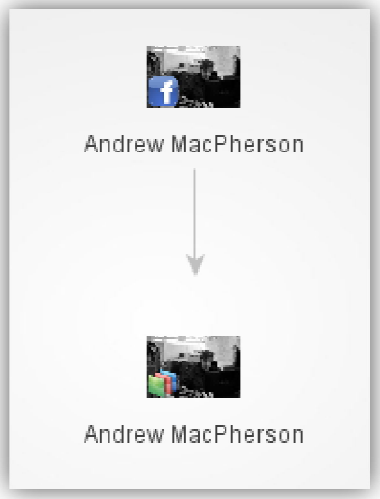
2.15.1 To Person from Profile

This transform uses the Facebook profile details to populate a person entity so that it can be used in conjunction with other transforms.

Additional Entities Required: None

2.15.2 To Profile Image

This transform creates an Image entity from the Facebook profile picture. This is useful for looking at exif and reverse image searching.

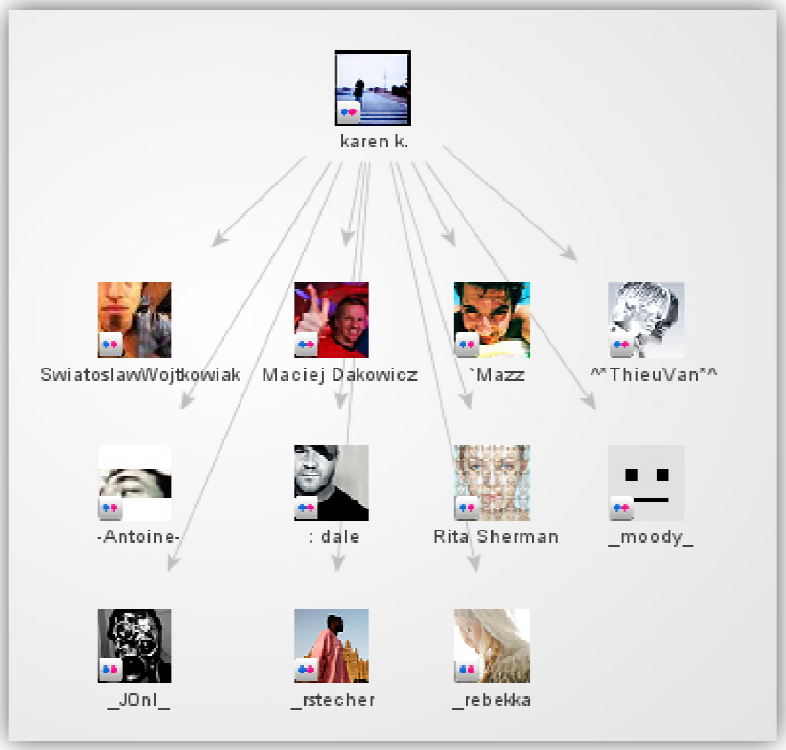


Additional Entities Required: Image

2.16 Affiliation Flickr

2.16.1 Get Friends

This transform enumerates friendship on the social network Flickr. Useful for identifying key nodes in friendship groups on this network.

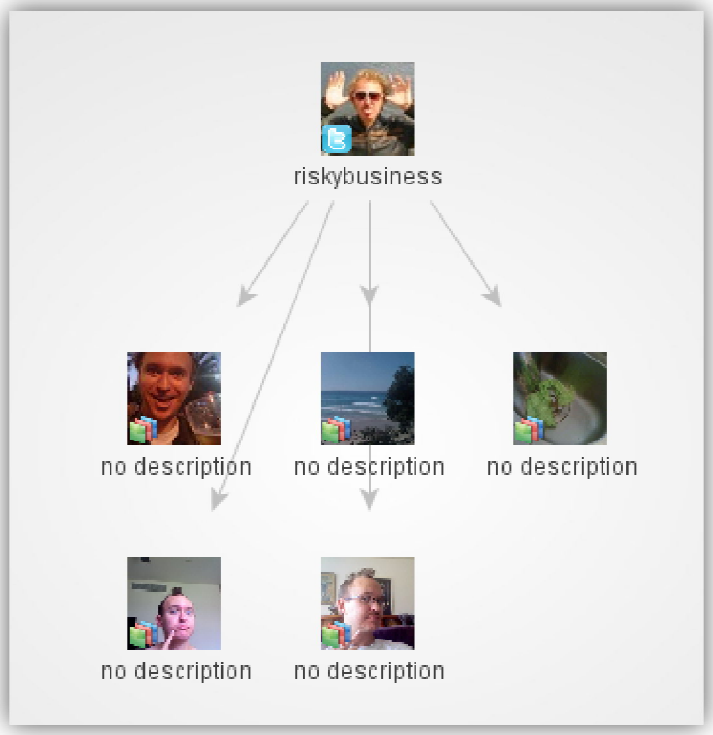


Additional Entities Required: None

2.17 Affiliation Twitter

2.17.1 To TwitPic Images

To TwitPic Images returns all Twitpic.com images based on a twitter affiliation. This is useful to the analyst for identifying exif information and reverse image searching.



Additional Entities Required: Image

2.17.2 To Twitter Profile Image

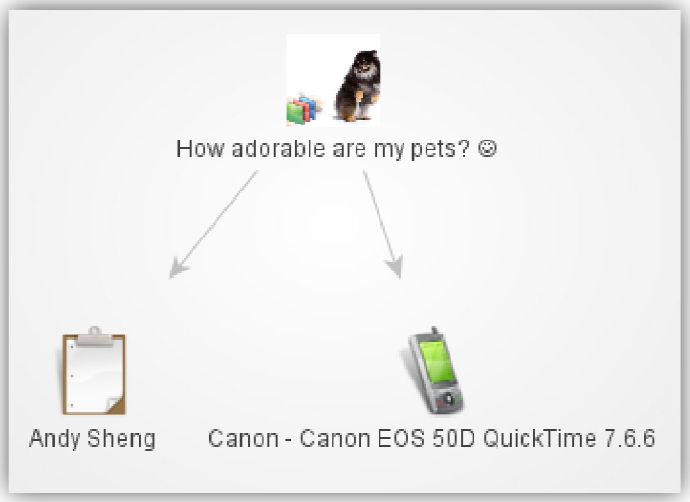
Simply returns the profile image from a twitter affiliation. This is useful for looking at exif information and reverse image searching.

Additional Entities Required: Image

2.18 Image

2.18.1 Get Exif Information from Image

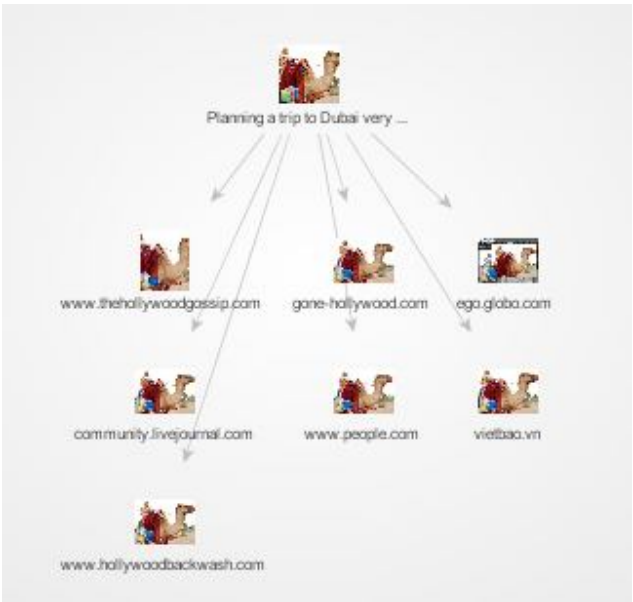
This transform attempts to identify various exif information contained within images such as device, author and GPS co-ordinates. Useful for identifying physical devices owned by a target as well as locations, software models, individuals and other information.



Additional Entities Required: Device, GPS

2.18.2 To Websites via Tineye

This transform uses the reverse image search engine Tineye to display websites where similar or the same image has been found. This is useful to the analyst to identify related sites



Additional Entities Required: Image