

[Purpose](#)

[Research Foundation](#)

[VT-MIS and Private API](#)

[Metadata Results Structure](#)

[Collection Process](#)

[Derived Hash Values and Account Types](#)

[Identifying Interesting Activity](#)

[Actor Account Characteristics](#)

[Uploader Behavior](#)

[Unique Filename Fixation](#)

[Case Studies of Actors](#)

[APT1 QBP2010](#)

[NetTraveler](#)

## Purpose

Virustotal provides two types of services, a free, public interface for users to submit files for analysis and a paid service that exposes significantly more data through a search interface about the files that were submitted. Apart from the data about the file submitted, Virustotal also provides metadata associated with the person who submitted the file. Research has concluded that it's possible to predict and identify victims, actors and other types of accounts based on their uploads to Virustotal over a period of time using the submission data associated with the files they upload. This paper will outline and discuss the process associated with coming to this conclusion and the understanding gained while working with the submission data.

## Research Foundation

In order to understand the results of the performed research, it's necessary to know more about the paid Virustotal services, how they present the submission history for a given file and how the data was collected since it was done through non-trivial means.

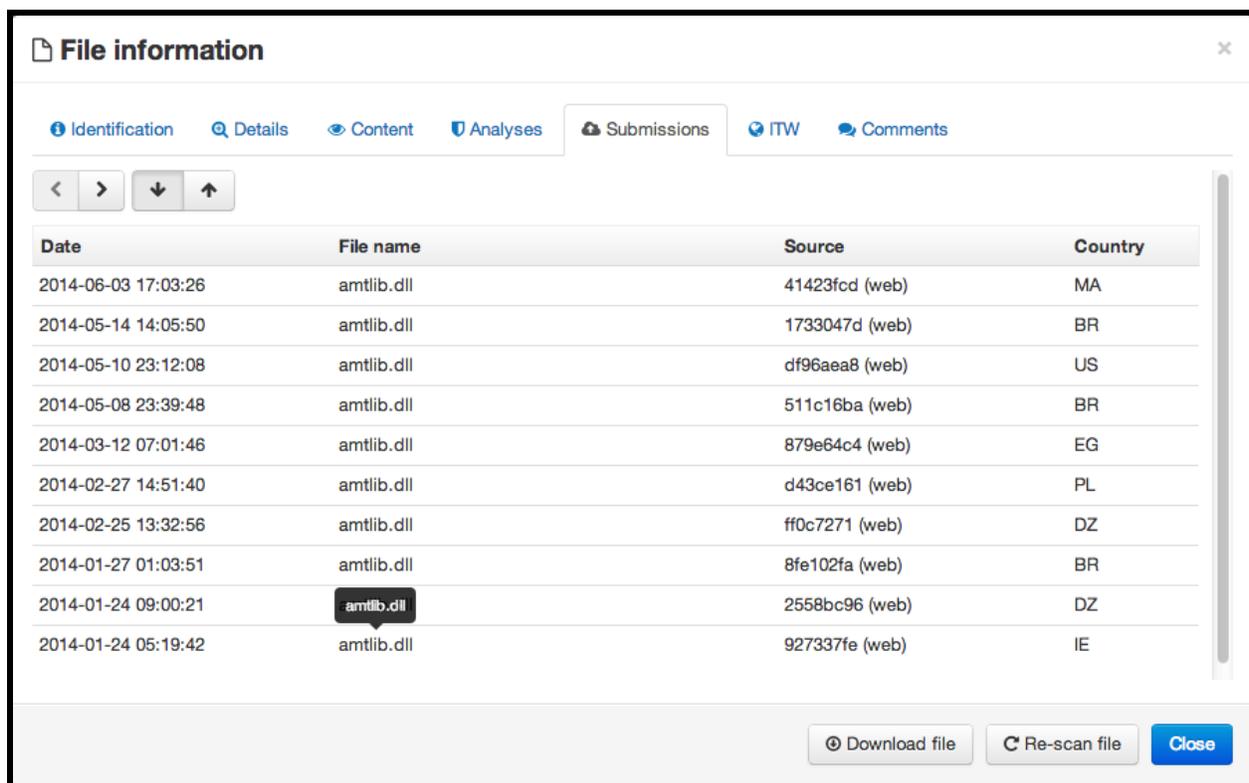
### **VT-MIS and Private API**

Virustotal, developed by Hispasec and purchased by Google in 2012, is a free service offered to the public to scan submitted files and URLs against multiple anti-virus scanning engines. Aside from the free interface, Virustotal also offers the Malware Intelligence Service (VT-MIS) and Private API, paid services that expose backend searching capabilities and programmatic interfaces to obtain stored data about uploaded files. Using the paid interface APIs and web portal, researchers can identify new malicious samples through a number of different filters and tags provided by the Virustotal developers. Additionally, researchers can "hunt" for new samples

by uploading YARA signatures that when triggered, will provide a notification back to the researcher.

## Metadata Results Structure

Contained within the web portal of the VT-MIS service are a number of metadata details about the particular file that was uploaded. This metadata is subject to changes and additions, but generally remains constant depending on the file type. Included in the metadata for each file is a section describing the submission process itself as shown in Figure-XX.



The screenshot shows a web interface titled "File information" with a close button in the top right. Below the title are several tabs: "Identification", "Details", "Content", "Analyses", "Submissions" (which is selected), "ITW", and "Comments". Below the tabs are navigation arrows: left, right, down, and up. The main content is a table with four columns: "Date", "File name", "Source", and "Country". The table contains ten rows of submission data. The file name "amtlib.dll" is highlighted in the second row of the table. At the bottom right of the interface are three buttons: "Download file", "Re-scan file", and "Close".

Date	File name	Source	Country
2014-06-03 17:03:26	amtlib.dll	41423fcd (web)	MA
2014-05-14 14:05:50	amtlib.dll	1733047d (web)	BR
2014-05-10 23:12:08	amtlib.dll	df96aea8 (web)	US
2014-05-08 23:39:48	amtlib.dll	511c16ba (web)	BR
2014-03-12 07:01:46	amtlib.dll	879e64c4 (web)	EG
2014-02-27 14:51:40	amtlib.dll	d43ce161 (web)	PL
2014-02-25 13:32:56	amtlib.dll	ff0c7271 (web)	DZ
2014-01-27 01:03:51	amtlib.dll	8fe102fa (web)	BR
2014-01-24 09:00:21	amtlib.dll	2558bc96 (web)	DZ
2014-01-24 05:19:42	amtlib.dll	927337fe (web)	IE

Figure-XX: Submission history for a file

This record is made up of four different sub-metadata pieces to include the date and time of submission, filename, country and unique hash derived from the method in which the file was uploaded. Each time the file is resubmitted for scanning, Virustotal will add to the submissions record with the corresponding metadata.

## Collection Process

Users who have purchased access to the VT-MIS service can also purchase access to the Virustotal Private API. This API allows developers to programmatically access most of the data a user would see within the VT-MIS web interface. However, when it comes to submission data, there is no way to access a given file's complete submission history through a documented API

interface. In order to collect the full submission history, web scraping must be done to extract the data presented on the VT-MIS website and save it to a local database.

Submission data within VT-MIS is structured inside of an HTML table that allows for sorting based on the submission date/time. By default, submissions are sorted by the most recent submissions first. On files with over ten submissions, pagination is employed to show ten records view. Through programmatic means, it's possible to recursively crawl and extract all submission history of a file into a structured data format that can be stored with results from the private API for local analysis. The resulting data structure makes the focus on who submitted the file and not the file itself.

Acquiring the submission history of a file is just one aspect of the collection. While VT-MIS shows the user the submitter hash, it's not possible to search the interface using that hash. This presents an issue in finding other files a given user account may have submitted. The only way to account for this lack of visibility is to process as many file hashes as possible in order to collect new or existing submitter hashes. This of course means the collection process is always bias to what file hashes are being requested and also means significant submissions for users could be missing. Regardless of the bias, it's possible to build up enough activity for submitter accounts to perform analysis on their activities.

## Derived Hash Values and Account Types

In order to preserve privacy within VT-MIS, Virustotal employs a one-way hashing algorithm that takes contextual details from the submission and some unknown value (likely a dynamic salt) to ensure the data being protected can not be guessed or brute-forced. The format of the one-way output is an eight character hash consisting only of hexadecimal values (0-9, A-F) and a source identifier of "web", "api", "email" or "community" (and rarely anti-virus companies) derived from how the file was uploaded to Virustotal.

By analyzing various submitter accounts and their types, it appears that the following rules are being applied by Virustotal:

<b>Account Type</b>	<b>Explanation</b>
Web	The user has submitted their file through the public Virustotal interface while not logged in to any account. The IP address from which the submission was made is appears to be the primary source of the hash.
API	The user has submitted their file via the public Virustotal API. The company/identity of the

	user appears to be the primary source for the hash.
Email	The user has submitted their file via email to the Virustotal email alias. The email address that sent the sample to Virustotal appears to be the primary source of the hash.
Community	The user has submitted their file through the public Virustotal interface while logged into their community account. The identity of the user appears to be the primary source of the hash.

Using the above-mentioned assumptions, the following caveats can be identified:

- Due to network address translation (NAT), Web accounts could represent an entire organization, not just one user.
- Due to the primary source for hashing, API and Community accounts could have different source countries despite having the same hash
- Due to email addresses not having a location, Email accounts would likely never contain any country data

Despite these caveats, value can still be gleaned from the submission data.

## Identifying Interesting Activity

Interesting is a relative term, but for this research, “interesting” meant any activity associated with a submitter account that appeared to be an “actor”. There was no prior work to identify these types of accounts based on their file uploads, so four different classification labels were created for an analyst to use during manual review as outlined in the table below.

<b>Class Type</b>	<b>Explanation</b>	<b>Rules</b>
Security	Users that appear to be anti-virus companies, security firms, independent researchers or security operation centers based inside of an organization	<ul style="list-style-type: none"> <li>• Submits files with names matching common patterns</li> <li>• Uploads hundreds of files on a daily basis and thousands overall</li> </ul>
Target	Users who submit files associated with active or past malicious campaigns	<ul style="list-style-type: none"> <li>• Uploads files with names likely used in an actual attack</li> </ul>

	with “real-world” file names not matching a specific pattern.	<ul style="list-style-type: none"> <li>• Upload amount varies, but does not exceed several thousand</li> <li>• Usually does not submit files with a common hash pattern name</li> </ul>
Actor	Users who submit files in order to identify how their malicious file is being detected by the multiple anti-virus companies.	<ul style="list-style-type: none"> <li>• Uploads a smaller amount of files</li> <li>• Performs “burst” uploads</li> <li>• May follow a non-conventional pattern on filenames</li> <li>• Payloads may be missing critical components or show signs of incremental changes</li> </ul>
General	Users that appear to be associated in a number of categories, but mostly upload benign information or files with a high number of associated submitters	<ul style="list-style-type: none"> <li>• Re-uploads files that have been scanned hundreds of times</li> <li>• Uploads a variety of benign and malicious samples</li> <li>• Payloads do not have a clear connection between each other</li> </ul>

These labels were applied by an analyst reviewing a particular submitter in order to establish a classified dataset that could be reviewed for patterns. Upon labeling several hundred accounts using the rules listed above, a few patterns emerged that a computer could use in order to identify likely actor accounts.

**Actor Account Characteristics**

The primary goal of an actor using Virustotal is to understand how their malicious code is being detected across one or more anti-virus solutions. Sometimes this is done in order to evade a specific antivirus, say one deployed inside the targeted environment, but other times done in order to not be detected at all. During the research, two methods were discovered to track and identify likely actor activity within Virustotal. The outlines below describe these methods.

**Uploader Behavior**

Given an attacker’s goal is to evade antivirus solutions, it’s assumed they will need to make minor changes to their code and re-upload their files for analysis. This behavior of uploading for analysis, making changes and then uploading again creates a pattern that can be identified using four different features. The table below outlines these features.

<b>Feature</b>	<b>Role</b>
Submits Duplicates	An account that submits duplicates is looking to understand if a file detection has changed over a period of time. Actors will

	re-submit the same file over and over in order to understand if their code needs to be adjusted. Likely times of performing this would be on malicious code currently being deployed in an operation
Submits Valid Filenames	Actors seem to favor pattern-based names for their uploaded files. It's rare to identify an actor using names that match common hashing algorithm patterns. A valid filename is that not matching a list of defined patterns.
Short Delta Between Submissions	Accounts that submit several files with short periods of idle behavior in between are suspicious. This sort of activity generally occurs when an account is attempting to evade antivirus detection.
Filename Pattern Re-use	Observed actor accounts have favored using patterns inside of filenames. Re-using or following a pattern in the filename can be a sign of an actor incrementing their versions or modifications as they attempt to evade antivirus.

By assigning scores to each one of the above mentioned features and reducing the activity period to a week, it's possible to derive a total score for a given accounts activity. This score value can then be used in conjunction with other features such as total number of uploads, country of origin and so on to improve the filtering of suspicious accounts. If implemented properly, it's possible to automate the discovery of likely actor activity for an analyst to review.

**Unique Filename Fixation**

One of the features described in the uploader behavior section was filename pattern re-use. This feature stated that actors would often re-use filenames for their malicious code, but assumed that actors activity would always be associated with one account. Research has identified actors who use Virustotal on a regular basis, but appear to come from different IP addresses every time they submit a file which means their behavior can't always be tracked. However, if the actors use unique filenames, it's possible to track their activity using the search provided by Virustotal.

Contained within VT-MIS is a powerful search interface where licensed users can search for files with a given filename. If analysts know an actor account always comes from China and uses the filename "HT.exe", then it's possible to perform a search for this activity with a high degree of success in finding actor accounts. Taking this concept a step further, it's possible to automate the searching and collection of results so that this can be done on a defined schedule.

**Case Studies of Actors**

Taking years of knowledge associated with Virustotal submitters and algorithms that used features defined in this paper, it was possible to identify several actor accounts within Virustotal associated with different countries.

## **APT1 QBP2010**

Starting in mid-August of 2012, actors likely associated with APT1 began uploading variations of their WebC2-QBP2010 malware to Virustotal. This behavior continued in varying degrees through the next several weeks where it ultimately ended in November of 2012. During this short testing period, actors appeared to focus on several different routines within their code including the main command and control thread, decoy document processing and registry persistence. On numerous occasions, actors were successful in severely reducing their detections on their malicious payloads before sending them out to target organizations.

Account activity showed a high operational tempo with limited oversight on operations. On one particular occasion, actors accidentally introduced a bug into their code that caused an error to occur when opening the decoy document. This malicious payload was sent to the target organization anyway despite the error. The next day, actors identified the mistake, fixed it, changed decoy documents and targeted the same organization again.

Additionally, actors appeared to demonstrate an increased operational security position over time. Many of the first testing files were uploaded from a single account and included command and control information. As time progressed, actors began to come from different accounts every time and replaced command and control data with placeholders as to not give away their infrastructure before conducting the operation.

In total, over 15 different accounts appeared to be used by the actors to upload over 100 different malicious samples. Data detailing the targeted organizations was obtained through private sources and offered a rare compliment to the testing data. Actors associated with APT1 did not appear to use Virustotal again.

## **NetTraveler**

Since 2009, actors associated with the NetTraveler campaigns have uploaded their files to Virustotal. Unlike most actors, NetTraveler doesn't follow the standard behavioral patterns and appears to look more like a target account than an actor. The discovery of this account was made through the filename patterns and more specifically, the files uploaded by the actors.

Files uploaded by NetTraveler generally come from infrastructure used for command and control of their malware and phishing operations. Much of their targeted is focused on Tibet and other political dissidents and as such, a lot of their uploads include phishing email and the malicious payloads used as attachments.

Through submission history analysis, it was possible to identify one situation where NetTraveler actors appeared to compromise a newspaper reporter's email account. They then downloaded several benign emails that included attachments, implanted the files with their malware and then re-sent the mail out to a recipient list that was not part of the original legitimate email.

It's unclear why actors involved in the NetTraveler group upload files to Virustotal, but in doing so, they themselves have become the best source of information when tracking the group. Actors associated with NetTraveler remain active within Virustotal and upload on a daily basis.