**Web Interface Portal & Security Threat Engine for REMnux (WIPSTER) User Guide**

1. Introduction
2. Installation
3. Administration Console
4. Malware Analysis
5. URL Analysis
6. Pastebin Search
7. Twitter Feed
8. Converting Strings

## Introduction

The Web Interface Portal & Security Threat Engine for REMnux (WIPSTER) was designed to be used as an automated submission platform for the Malware Analysis distribution, REMnux (<http://zeltser.com/remnux/>).

Utilizing various plugins and open source tools, it allows for both static and dynamic analysis of suspected malware and malicious links with one submission. As development continues, additional features have been added such as the ability to automatically search and record Pastebin for select keywords, maintain a stream of Twitter posts matching a selected syntax, or convert an inputted string to various computer- and user-friendly formats.

## Installation

WIPSTER is included in REMnux v5, and can be installed by simply running the following from the command line:

/usr/local/sbin/install-wipster

On launch, the installation script will prompt you for your version of REMnux, install the necessary requirements, ask whether or not you wish to automatically check Pastebin for specific keywords every hour, and finally present you with the administration console where you can change your settings and input your various APIs.

If you are running REMnux v4, installation can be achieved by running the following from the REMnux command line:

cd ~

wget https://raw.githubusercontent.com/TheDr1ver/WIPSTER/master/install.sh --no-check-certificate

chmod +x ./install.sh

sudo ./install.sh

**Note:** If installing in REMnux v4, be sure that your source repositories have been modified to include the old-archive URLs **before** running the WIPSTER installation script

sudo sed –i –e ‘s/archive.ubuntu.com\|security.ubuntu.com/old-releases.ubuntu.com/g’ /etc/apt/sources.list

sudo sed –i –e ‘s/us.old-releases.ubuntu.com\|security.ubuntu.com/old-releases.ubuntu.com/g’ /etc/apt/sources.list

sudo apt-get update

**Note:** Users of NoScript should allow all scripts to run from the REMnux IP (or localhost), otherwise forwarding and certain CSS tweaks will not work properly.

## Administration Console

After WIPSTER has been successfully installed, you will be presented with the Administration Console (localhost/admin.php) inside your REMnux distribution. There you will be able to set the following options:

* REMnux Settings
  + Select version 4 or version 5 (changing this is a bad idea at the time of this writing)
* MASTIFF Settings
  + Location of the MASTIFF config file
    - The location of your MASTIFF config file (stores your VirusTotal API for MASTIFF, among other settings)
    - REMnux v4 Default: /usr/local/etc/mastiff.conf
    - REMnux v5 Default: /usr/local/mastiff/mastiff.conf
  + Location of the MASTIFF python script
    - The actual script that runs MASTIFF in the background
    - REMnux v4 Default: /usr/local/bin/mas.py
    - REMnux v5 Default: /usr/local/mastiff/mas.py
* TRiD Settings
  + Location of TRiD
    - REMnux v4 Default: /usr/local/lib/triddefs.trd
    - REMnux v5 Default: /usr/local/TrID/triddefs.trd
* Malwr.com Settings
  + Malwr checkbox – check if you wish to allow the option for submitting to Malwr.com
  + Malwr.com API – get it by signing up at <https://malwr.com/account/signup/> - View your Profile and click on the API tab
* ThreatAnalyzer Settings (more info: <http://www.threattracksecurity.com/> )
  + ThreatAnalyzer checkbox – check if you wish to use a locally installed copy of ThreatAnalyzer
  + ThreatAnalyzer API
    - Obtained from your ThreatAnalyzer administration page – can only be used by a ThreatAnalyer administrative account
  + Base IP
    - The IP of your ThreatAnalyzer server
  + Submission Priority
    - Submission priority of analysis jobs submitted using WIPSTER
  + Sandbox Group option
    - custom – submit to one specific sandbox based on MAC address
    - for\_any\_group – submit to any sandbox in a particular group
    - for\_all\_group – submit to all sandboxes in a particular group
  + Sandbox Group for Submission
    - Group ID for submission (if for\_any\_group or for\_all\_group selected above)
  + Custom Action Name
    - Custom action to run after analysis is complete
    - default is ActionAfterAnalysis
  + Custom Action Value
    - Custom action value
    - Default is revert
  + Reanalyze previously submitted files checkbox – check if you wish to run samples previously seen by your ThreatAnalyzer instance
* Anubis Settings
  + Username
  + Password
  + Sign up for Anubis for free at <https://anubis.iseclab.org/?action=register>
* Open Source URL Checkers
  + MyWOT API
    - Request a free API key at <https://www.mywot.com/en/signup?destination=profile/api>
  + VirusTotal API
    - Request a free API key at <https://www.virustotal.com/en/#dlg-join>
  + Google Safe Browsing API
    - Request a free API key at <https://www.google.com/safebrowsing/key_signup.html>
* Pastebin Checker Settings
  + Google Custom Search Engine API Key
  + GCSE Signature Key
  + GCSE ID
  + GCSE Query
    - The query automatically run by the Pasebin Checker plugin
  + User Agent for Pastebin Checker
    - Default is Mozilla 4.0
  + More info can be found in the Pastebin Checker section
* Twitter Settings
  + Request a free Twitter API Key at <https://dev.twitter.com>
  + API Key
  + Access Token
  + Query
    - Search query to run in the Twitter plugin
    - Default: #0Day -RT filter:links
  + Consumer Secret
  + Access Token Secret
* Backup WIPSTER
  + Save all settings, samples, etc. Basically everything in /var/www/
* Download Configs
  + save a copy of your configurations - /var/www/admin/admin.db
* Update WIPSTER
  + Download the most recent files from the GitHub repository, leaving your configurations, databases, ticketgen.php page, and existing samples intact

## Malware Analysis

Currently, Malware Analysis has been broken down into three sections as of this writing:

1. View MASTIFF Results Dashboard
2. Submit files to MASTIFF
3. Search MASTIFF Database

The results dashboard shows the most recent 25 analyses submitted to WIPSTER, Submit files to MASTIFF is where files can be submitted for analysis, and Search MASTIFF Database allows for searching the previously submitted files either by MD5 or Ticket number.

### Submission:

Here you choose your file to submit to MASTIFF.

Any filetype is permitted, but if you are submitting to ThreatAnalyzer or Malwr.com, it is best to submit files with their original extension.

Ticket numbers are required for cross-referencing, and currently must be between 7 and 12 digits long.

Notes can be added to the database in the Notes field, and the ThreatAnalyzer and Malwr.com checkboxes can be used if their respective plugins are activated under the Administration Console.

### Results Page:

After submission, a brief summary of debugger information will be presented, and your browser will automatically forward to the results page. The header of the results page includes buttons for running No-More-XOR against the file in question, viewing a full summary of all static results in a single plain text page, and submitting the file to ThreatAnalyzer if the plugin is activated on the Administration Console.

The results page is broken down into the following sections:

* Summary
  + Samples Analyzed
    - Names, tickets, and links to all files matching this MD5
  + Related Tickets
    - All tickets submitted using this particular MD5
  + Fuzzy (SSDeep) Hash results
    - Results of SSDeep against all files uploaded to WIPSTER, used to determine similarity between samples
  + Zip Contents (if file uploaded is an archive)
    - Links to each MD5 summary of all files uploaded in the zip archive
  + PCAP Analysis (if file uploaded is a pcap)
    - Attempts to determine if the PCAP is web or email traffic
    - If email traffic is determined, and base64 code is found, attempts to extract any attachments by converting the base64 code to a binary and provide a link for downloading
  + Strings
    - Searches for strings containing http:// and strings containing Program
  + De-XOR
    - If De-XOR has been run, Searches for de-xor’d strings containing http:// and de-xor’d strings containing Program
* OpenSource Checks
  + VirusTotal Results
  + Malwr.com Results (if Malwr.com plugin is checked on Administration Console)
    - Provides a link, allowing submission to Malwr.com
    - If submission is accepted, checks the unique ID of the analysis to provide the status and a link to the results
    - **NOTE:** at this time, it doesn’t appear malwr.com is properly accepting submissions… One particular analysis I sent has been stuck at ‘processing’ for several days.
  + Anubis Results
    - Provides a link to automatically submit samples to the Anubis sandbox, using the username and password provided in the WIPSTER Administration Console
    - If TCP connections are found in Anubis, the results will be listed here
* Files produced by MASTIFF
  + Shows a list of all files created and extracted by running the sample through MASTIFF
* Web Searches
  + Link to Google Search (since iframes are blocked by Big G) of the MD5
  + iFrame for DuckDuckGo search of the MD5
  + iFrame for StartPage search of the MD5
  + iFrame for ThreatExpert search of the MD5

### Submit to ThreatAnalyzer / ThreatAnalyzer Details

If the ThreatAnalyzer plugin is enabled on the WIPSTER Administration Console, you may integrate WIPSTER with your network-accessible installation of ThreatAnalyzer. This button will read either **Submit to ThreatAnalyzer** if the sample was not found in the existing ThreatAnalyzer database, or **ThreatAnalyzer Details** if it was.

Clicking **Submit to ThreatAnalyzer** will send the sample to your ThreatAnalyzer instance as per the configurations on the WIPSTER Administration Console.

Clicking **ThreatAnalyzer Details** will replace the static analysis summary with a brief ThreatAnalyzer summary, including:

* Analysis number (linked)
* Sandbox name
* Analysis status
* Filename analyzed
* MD5 (linked) of sample
* Time submitted
* Link to download a PCAP of the analysis network traffic
* Risk summary & reasons for determined risk
* Network connections:
  + Domains
  + IPs
  + Commands (GET/POST) sent

### View Plain Text

Clicking the **View Plain Text** button checks the following tools/sources, and provides a summarized report of static and dynamic analysis found:

* Ticket # with subject line templates
* Reason for ticket alert: blank
* DNS callout (populated per Anubis and ThreatAnalyzer results)
* IP callout (populated per Anubis and ThreatAnalyezr results)
* Block/Blackhole: blank
* Filename
* MD5
* Filesize
* File Path: blank
* Filetype (per MASTIFF)
* CVE: blank
* Link(s): blank
* Caught by AV (Yes/No) – Results grabbed per MASTIFF analysis, virustotal API required in mastiff config file
  + VT – pos/total
  + SEP – signature
  + MSE – signature
* Email info: blank
* Summary section: blank
* Remediation section: generic templates
* Notes (fields will appear when positive results were found
  + VirusTotal results (per vt api in mastiff configs)
  + Metadata (from mastiff)
  + EXIF Data (using exiftool)
  + Filetype info (per TrID)
  + Yara results (per mastiff)
  + PeePDF Results (if filetype is PDF, per peepdf)
  + PDFID results (per MASTIFF)
  + PEFrame Results (if filetype is EXE, per peframe)
  + PEScanner Results (per mastiff)
  + OLE Scanner Results (per mastiff)
  + De-Xor’d Strings (if nomorexor was run)
  + Strings (per mastiff)

**Note:** Clicking the **View in Plain Text** button uses PHP Session variables from whatever the last MD5 page you viewed was. Therefore, if you let your session expire, the page won’t work right. Also, if you open multiple MD5 summaries at once, clicking this button will use the last variables that were set – so you may have to refresh the page of the MD5 you wish to analyze before clicking this button.

## URL Analysis

URL Analysis is done through the URL Research Page (/urlResearch.php). Options for URL Research at this time include:

* VirusTotal
* MalwareDomains.com
* Google SafeBrowsing
* Web of Trust (MyWOT)
* Wepawet
* Sucuri
* IPVoid
* RexSwain
* URLQuery
* Quttera
* DuckDuckGo
* StartPage

In a red box at the top of the page, the alerts section will show:

* If the URL has been previously submitted, and what ticket number it correlates to
* If the response headers are 200/404/301, etc. If the headers are a redirect, the submission box will automatically repopulate with the URL for the redirect
* VirusTotal results summary
* MyWOT results summary
* If Sucuri did not find the website benign
* MalwareDomains.com results summary
* Wepawet results summary
* Google SafeBrowsing results summary

**NOTE:** At this time only HTTP websites can be submitted to WIPSTER. HTTPS sites will automatically be converted to HTTP before analysis.

## Pastebin Search

### Setup

Searching Pastebin and using the automated Pastebin search utility requires a little more preparation to set up properly:

1. Visit <https://www.google.com/cse/all> and click **Add** to design your custom search engine.
2. Name the engine whatever you’d like
3. Under Details click **Search Engine ID** – this is your WIPSTER **Google Custom Search Engine ID**
4. Under sites to search add whatever pastebin sites you would like to scrape for data. Some suggestions are:
   1. leakedin.com/\*
   2. slexy.org/\*
   3. paste.frubar.net/\*
   4. pastie.org/\*
   5. pastebin.com/\*
5. Under Search features-> advanced, delete your existing keys, and add the following   
   Sort By->Label pairs
   1. [blank] -> Relevance
   2. date->Date
6. Websearch Settings
   1. Query Parameter Name = q
7. Websearch Restricts
   1. Sort By = date
8. Go to <https://console.developers.google.com/> and click on your new Custom Search Engine
   1. Under APIs & Auth -> Credentials
      1. Public Access API Key – write this down for the WIPSTER **Google Custom Search Engine API Key** on the Administration Console.
      2. OAuth Client Secret – write this down for the WIPSTER **Google Custom Search Engine Signature Key**

**NOTE:** Going back, I couldn’t figure out exactly where the Signature Key comes from, though I’m pretty sure it’s the OAuth Client Secret. An example URL Request made by the program looks like this:

https://www.googleapis.com/customsearch/v1element?key='**.**$gcseKey**.**'&rsz=filtered\_cse&num=10&hl=en&prettyPrint=false&source=gcsc&gss=.com&sig='**.**$gcseSig**.**'&cx='**.**$gcseCx**.**'&q='**.**$query**.**'&start='**.**$page**.**'&sort=date&googlehost=www.google.com&callback=google.search.Search.apiary13655&nocache=1395785766092

$gcseSig is where the signature key is called, so if you’re having problems with the custom search working, this is probably where it’s originating…

## Twitter

### Setup

You will have to build an app at <https://apps.twitter.com/>

Once built, under API Keys, you can access your API Key, Access token, Access token secret, and consumer (API) secret

After setup, a twitter feed for your search query will appear in your WIPSTER footer. By default, it searches for #0Day, only tweets with links, and ignoring re-tweets.

## Converting Strings

The converting strings page allows for converting elements to and from the following formats:

* Plain Text (ASCII)
* Base64
* URL Encoded
* Decimal
* HTML Entity
* Binary
* ROT 13
* Hexadecimal

And gets the following hashes for ASCII strings fed to the page:

* MD5
* SHA1
* SHA256
* SHA512