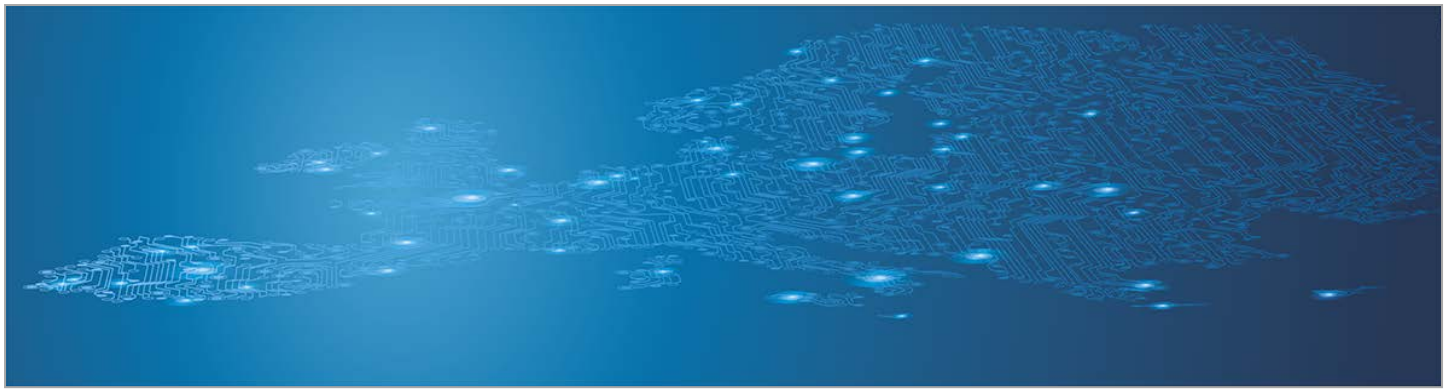

Identifying and handling cybercrime traces

Handbook, Document for trainers

September 2013

European Union Agency for Network and Information Security

www.enisa.europa.eu

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its Member States, the private sector and Europe's citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists EU Member States in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in EU Member States by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Authors

This document was created by the CERT capability team at ENISA in consultation with:

Don Stikvoort, Michael Potter and Alan Thomas Robinson from S-CURE, The Netherlands, Mirosław Maj, Tomasz Chlebowski, Paweł Wężgowiec from ComCERT, Poland, Przemysław Skowron from Poland, Roeland Reijers from Rubicon Projects, The Netherlands and Mirko Wollenberg from DFN-CERT Services, Germany.

Contact

For contacting the authors please use CERT-Relations@enisa.europa.eu

For media enquiries about this document, please use press@enisa.europa.eu.

Acknowledgements

ENISA wants to thank all institutions and persons who contributed to this document. A special 'Thank You' goes to the following contributors:

- Robin Ruefle from CERT Coordination Center, United States, Toomas Lepik from CERT-EE, Estonia, Thomas Lima from CERT.PT, Portugal, Krystian Kochanowski and Adam Ziaja from ComCERT SA, Poland, Vincent Danjen from Interpol, Andrew Cormack from JANET, United Kingdom, Katrina Sataki from NIC.LV, Latvia, Anna-Maria Talihärm, Estonia, Jerzy Kosiński from Police Academy, Poland, Jim Buddin from TERENA, The Netherlands.

Legal notice

Copyright Notice

ISBN 978-92-79-00077-5 doi:10.2788/14231

Table of Contents

Main Objective	The exercise consists of 3 components: finding relevant information related to cybercrime in social media channels (based on Twitter examples), finding relevant information on IRC channels and analysing legal aspects of Internet monitoring activities related to cybercrime identification. The main objective is to teach trainees how to set up the basic system for continuous monitoring and alerting of various sources of information in terms of effective detection and warning for their constituencies based on the content.	
Targeted Audience	CERT staff involved in the process of incident handling, especially those responsible for detection of new threats related directly to the CERT customers. As the exercise includes code development tasks, trainees should be able to code (especially in script languages – e.g. bash, php) or at least have a general capability of code execution recognition by reading the code.	
Total Duration	~ 7 hours	
Time Schedule	**Introduction to the exercise**	0.5 hour
	Task 1: Social Media (based on Twitter example)	2.5 hours
	Task 2: IRC channels	1 hour
	Task 3: Multiple online sources for finding relevant information	1.5 hour
	Task 4: Legal aspects of Internet monitoring services	1 hour
	Summary	0.5 hour
Frequency	You are advised to organise the exercise at least once a year. Even assuming that the main task, which is the development of the basic monitoring system based on the content, can be done after conducting the first exercise, the big value of regular exercise execution is the identification of new important sources of information.	

1 General Description

This exercise consists of three components. The first two components are the tasks for collecting all possible incident-related information with a special focus on information that is specific to various sources like social media and IRC channels. Very often this information is not IP-based information, which is a regular source of relevant information for CERTs. More and more relevant information is content-specific. Thus, working with the constituency requires a better understanding of their technical environment as well as methods of attacks on technical objects. For example, if the CERT provides services for a particular organisation which is an owner of the 'ABC123' system and the name of this system is specific and unique, then the CERT needs to start active network monitoring of all information related to such system. There are already many instances of the successful use of social media in tracking criminals. These include:

- Two men were identified as criminals who attacked (with the DDoS attack) Amazon, EBay and Priceline. They were bragging about this fact on an online hackers' forum. They were very active on the forum and shared a lot of information about various attacks and stolen credit cards[1]
- Hackers discussed break-in activity into the Sony PlayStation Network and the fact of credit card numbers possession on an underground Internet forum.

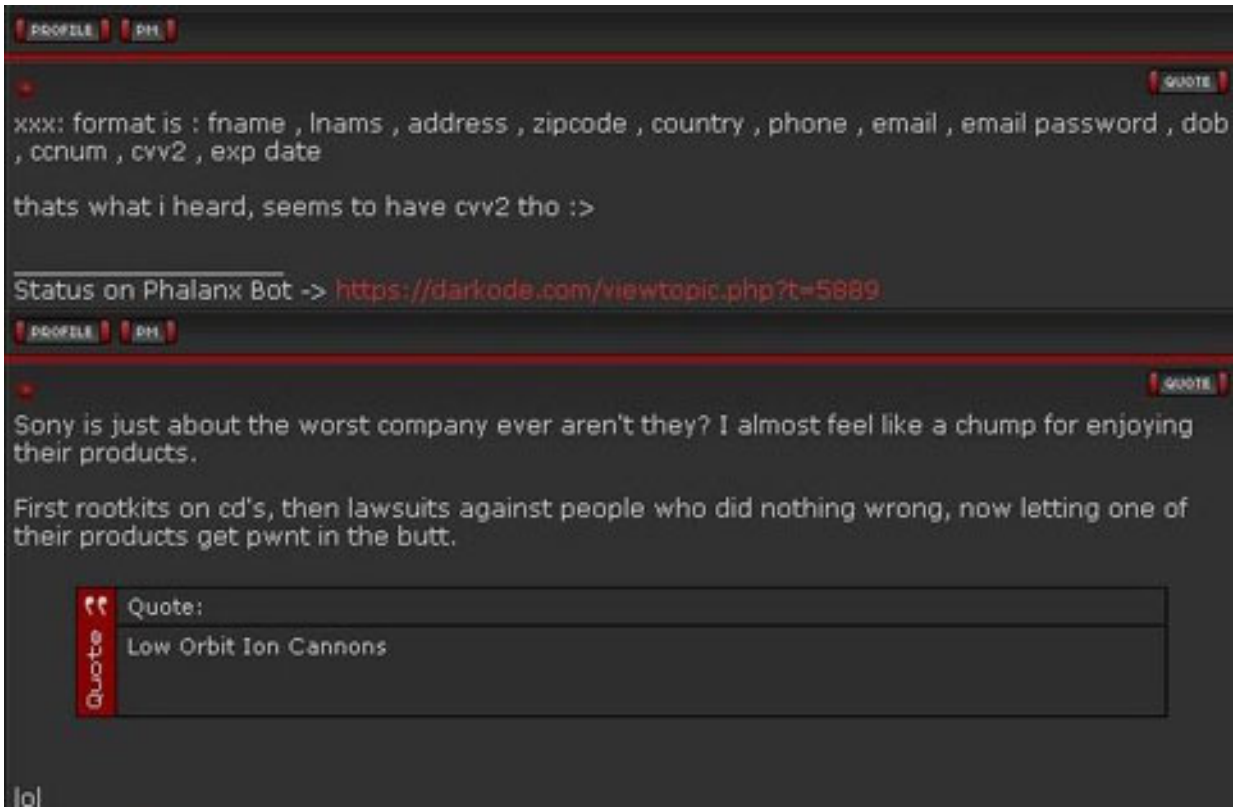

Figure 1: Hacker forum screenshot presenting discussion about the types of data hackers stole from Sony[2]

The first two subtasks of Task 1 (*Twitter accounts* and *keywords*) are universal and can be used as a template for the introduction work in Task 2 as well (except that in Task 2 we are talking not about Twitter accounts but about IRC channels). The following tools will also be used in the exercises in Task 3:

[1] *More:* http://arstechnica.com/security/2012/07/hacking-duo-charged-for-amazon-ddos/
[2] *Source: New York Times online service:* http://bits.blogs.nytimes.com/2011/04/28/hackers-claim-to-have-playstation-users-card-data/?_r=0

- Topsy.com service;
- NameChk.com service;
- Maltego (*http://www.paterva.com/web6/products/maltego.php*).

While performing Task 4 – Legal aspects of Internet monitoring services, trainees will learn about legal aspects of Internet monitoring activities. It is obvious that trainees come from different countries with different legal systems, but some general rules related to this topic can be taught. The example which will be presented in this exercise will be based upon national legislation and cover areas from Personal Data Protection Law.[3]

[3] *The Personal Data Protection Law in the EU Member States is based on the same directive – 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications). Its content (in official EU languages) is available at: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0058:EN:NOT*

2 Introduction

When preparing and conducting this exercise the trainer will focus on the following key factors:

- The CERT is responsible for a specific constituency;
- Part of the CERT services is network monitoring for active detection.[4]

The area of their interest should be various information sources, where they are interested in detecting a potential incident, not only from the technical point of view, but also from the point of view of identifying attackers. The team should also take into consideration all legal aspects related to the performance of their work according to existing law in their country or in the countries where they operate. Special attention should be paid to these activities, which are related to the regular collection and usage of data which could be treated as personal data. Important issues are: confidentiality of the communication, technical protection of the stored data, collection of sensitive data, attribution and handing over the assumptions made. It is also important that use of publicly available systems for content monitoring be limited to a minimum. The main reason is to avoid as far as possible the potential discovery of monitoring activity by potential cybercriminals.

The main schema for tasks related to identifying cybercrime traces is:

Figure 2: The general model for performing the exercise tasks[5]

There are two main open sources of open-source intelligence (OSINT) which will be discussed and analysed during the exercises:

- Social media (based on the Twitter example);
- Internet chat forums.

Additionally, there are further methods of collecting information from the Internet by using the services available online or after installing the required application on a computer.

[4] *Classic IP threats monitoring is not the main task during this exercise*
[5] *Abbreviations used in the figure: NM – network monitoring, IH – Incident Handling*

3 Task 1 – Social Media (based on Twitter example)

The possibility of social media monitoring will be analysed using the example of Twitter, which seems to be the most attractive source of potentially relevant information.

Trainees should be divided into groups of 3–4 people. It is important to have at least one person with programming skills in each group (bash/script languages are good enough and are preferable). Then trainees start to work in the groups.

3.1 Subtask 1 – Twitter accounts

The *first subtask* for trainees is to discuss and determine Twitter user accounts which, in their opinion, could become significant sources of relevant information. They can call their favourite Twitter accounts as well as carrying out Internet research during the exercise and collecting new favourites. The trainer should provide guidance for selection, such as:

- their main area of interest should be Twitter users' channels;
- there are some words which could be helpful in finding relevant channels, e.g.: 'anon', 'tango down', 'ops', 'corrupt', 'Cr3w', 'cyberwars'[6];
- their geographical location does matter, e.g. 'AnonInPoland' user channel;
- some periodic actions/operations can bring relevant information, e.g. '#OpUSA';
- trainees should focus not only on channels related to the particular groups. Some information channels, which specialise in monitoring these groups, are good intermediates.

At the end of this task groups should present their proposals of Twitter accounts. This short presentation (in the form of a simple list) should be followed by a short discussion about the quality and usefulness of the proposals. Groups should also propose their hackers' slang terms which can be used by trainees in their network investigations.

3.2 Subtask 2 – Keywords

After completing Subtask 1 groups should receive the *second subtask*. This subtask is to develop the list of keywords which will be used for monitoring and detection. In practice, when such services are provided, there are two sources of keywords:

- the set provided by the constituency representative. This type is usually very organisation-oriented and very often it refers to very specific systems of organisations' representatives like system names, particular persons' names, etc. On one hand this is very helpful as system owners are the best sources of relevant information, but on the other hand these keywords are impractical in terms of their existence in the underground sources of information and language used by criminals;
- the set developed by CERT members. This set is usually more practical in terms of the keywords' existence in the underground. It should be a natural addendum to the set provided by the constituency representative.

Good examples of keywords are:

- name of particular organisation (rather colloquial name than official name, e.g. 'ENISA' but not 'European Union Agency for Network and Information Security');
- English name of local name, e.g. translation into English from local language, like 'agency' (not 'agencja' in Polish);

[6] *Sample terms*

- even if we do not focus on IP addresses, it is good to have them in our set and treated as the text string;
- domain name of the monitored organisation or part of the constituency, e.g. 'enisa.europa.eu' or 'europa.eu';
- word usually used when information about successful attack is issued, e.g.: 'tango down', 'p0wned', and 'hacked'. If local language words are also often used in such a situation, they should be added to the set.

At the end of this task groups should present their proposals of keywords. This short presentation (in the form of a simple list) should be followed by a short discussion about the quality and usefulness of the proposals.

3.3 Subtask 3 – code development

The next, *third subtask* is to develop the code of the monitoring script. The main task of the code is to monitor chosen users' channels and alert whenever condition of monitoring is met.

The modules of the script could be the following:

Figure 3: Modules of the Twitter monitoring code

For the code development purposes the common set examples for all groups from the subtasks can be used.

The code development is optional. There are two solutions for performing this task:

1. As a minimum, the trainees change the keywords in the code for their own purpose and the best choice of keywords, according to their opinions.
2. As a maximum trainees develop their own code. In such case the code should include all modules indicated in *Figure 3: Modules of the Twitter monitoring code*.

The script can be found on Virtual Machine at: */home/enisa/enisa/monitoring/*.

A few screenshots from the script and the result of its processing are presented below.

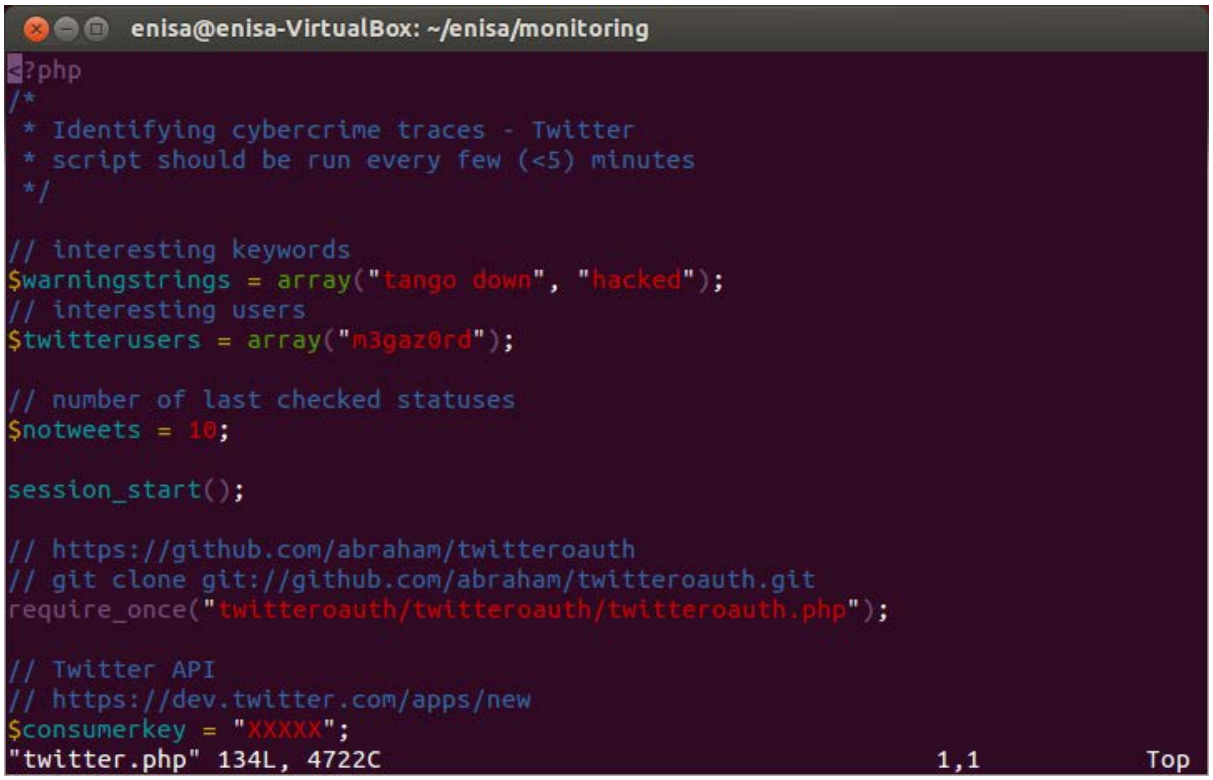

Figure 4: Screenshot of the script for Twitter monitoring: script code

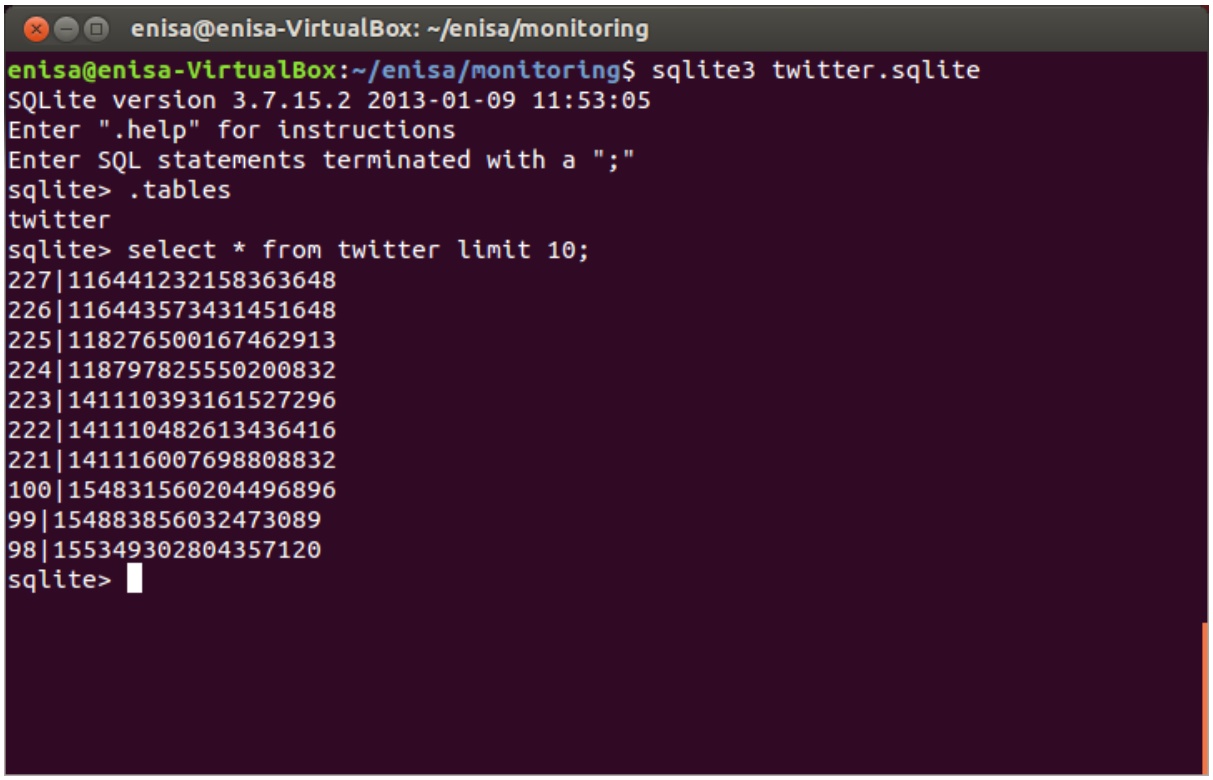

Figure 5: Screenshot from the script for Twitter monitoring: the tweets' ID numbers

The code example and its functional description is presented in Appendix 2 – The code example 1 for network monitoring (Twitter).

An example of the script is given below. In this example the keyword which was matched is 'hacked' and monitored Twitter account is 'AnonOpsLegion'.

WARNING ALERTS:

2013-07-03 20:05:18

https://twitter.com/AnonOpsLegion/status/352488234414112769

Muslim Brotherhood spokesman says all his social media feeds are hacked ll #Egypt

http://t.co/9b9fEe8MUv

For statistical presentation of the search results another script can be used (see: Appendix 3 – The code example for visual presentation of the tweets searching).

The graphical output from the script execution for 'ENISA' keyword is as below:

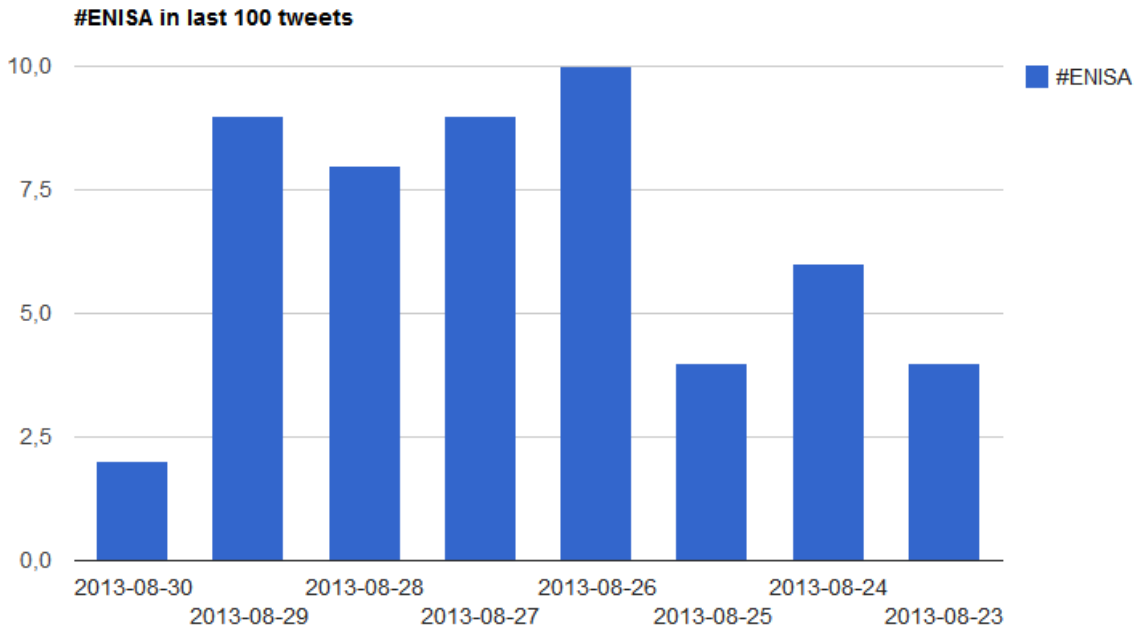

Figure 6: The graphical representation of the 'ENISA' keyword

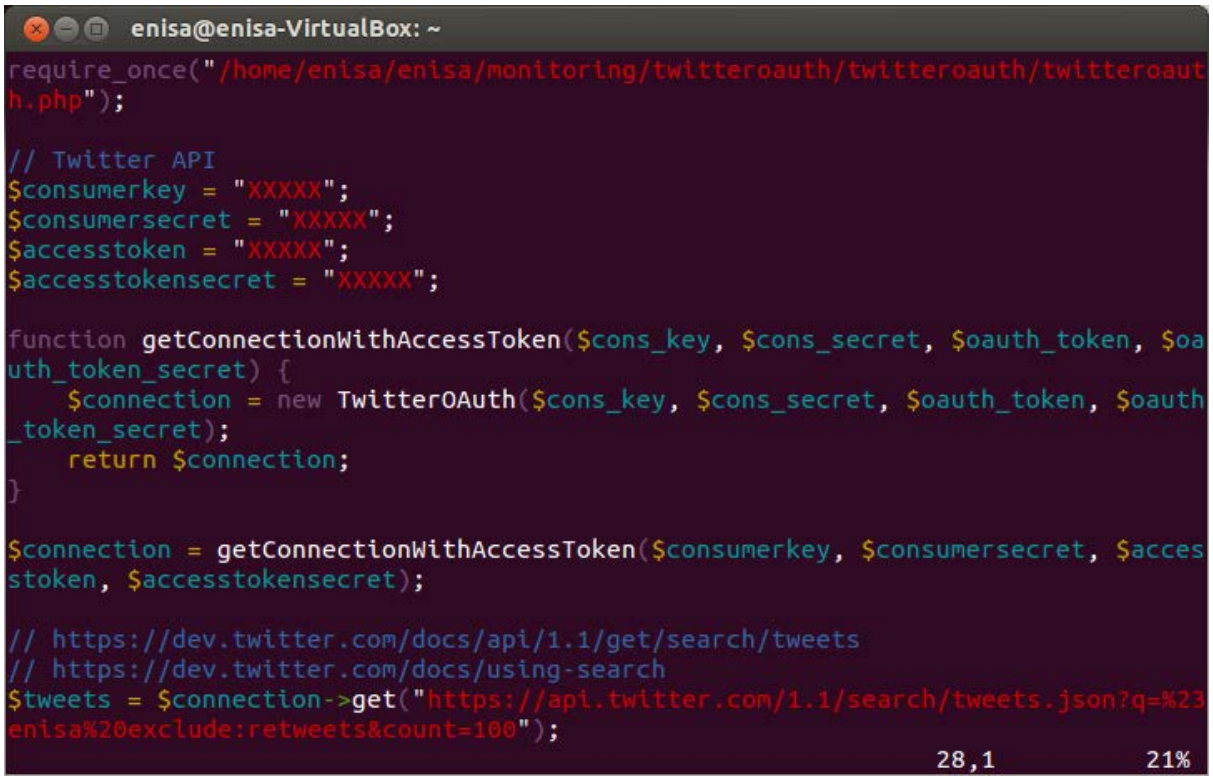

Figure 7: Screenshot from the script for graphical representation of the Twitter search results

3.4 Conclusion of Task 1

At the end of this task the trainer leads a wrap-up session. During the session trainees discuss:

- effectiveness of the search of Twitter channel;
- legal considerations related to this kind of search;
- their own experiences and ideas for effective monitoring of social media;
- the most interesting keywords (including hackers' slang words) for effective search;
- examples on how the graphical representation for Twitter monitoring solution worked.

4 Task 2 – IRC channels

During this task trainees will improve their skills of monitoring IRC channels. Automating this kind of work is probably one of the most difficult tasks for security professionals.

The *first subtask* will be to analyse security aspects of IRC channels monitoring. The most dangerous aspects are the possibility of identification of the person or organisation carrying out the monitoring, or the discovery of the fact that the presence on a channel is only for monitoring and discovering criminal activity purposes. Such identification can provoke attacks against an investigating party.

Trainees should develop their own ideas on how to challenge the above problems. This part of the exercise should not be treated as the most important aspect and trainees should not spend a significant amount of time on it.

The main ideas can be:

- to use anonymity in the network connection (e.g. with TOR service). The IRC channel can be reached anonymously by executing the 'torify' command which is a part of the 'tor' package (Ubuntu and Debian distributions). If we want for example to use irssi client the following command should be executed: *torify irssi;*
- to periodically make a 'human action' on the channel in order to be recognised as a trusted party;
- to periodically share potentially valuable information (from the criminal's perspective). This information should not bring a real value and for example could be re-published from other public sources.

The *second subtask* will be to develop the script which will alert investigators about a relevant IRC conversation. The assumption is that investigators have a secure IRC channel with functionality in place to reduce the possibility of their detection and identification. Their main goal is to develop a solution which will search IRC content logs, match them with keywords and finally alert the investigators via email message. Regarding the keywords, the rules for their setting up are exactly the same as those related to the social media channels.

The first part of the script should match the content of the logs with the keywords set (see Appendix 4 – The code example for IRC monitoring).

The script can be found on Virtual Machine at: */home/enisa/enisa/monitoring/*.

A few screenshots from the script and the result of its processing are presented below.

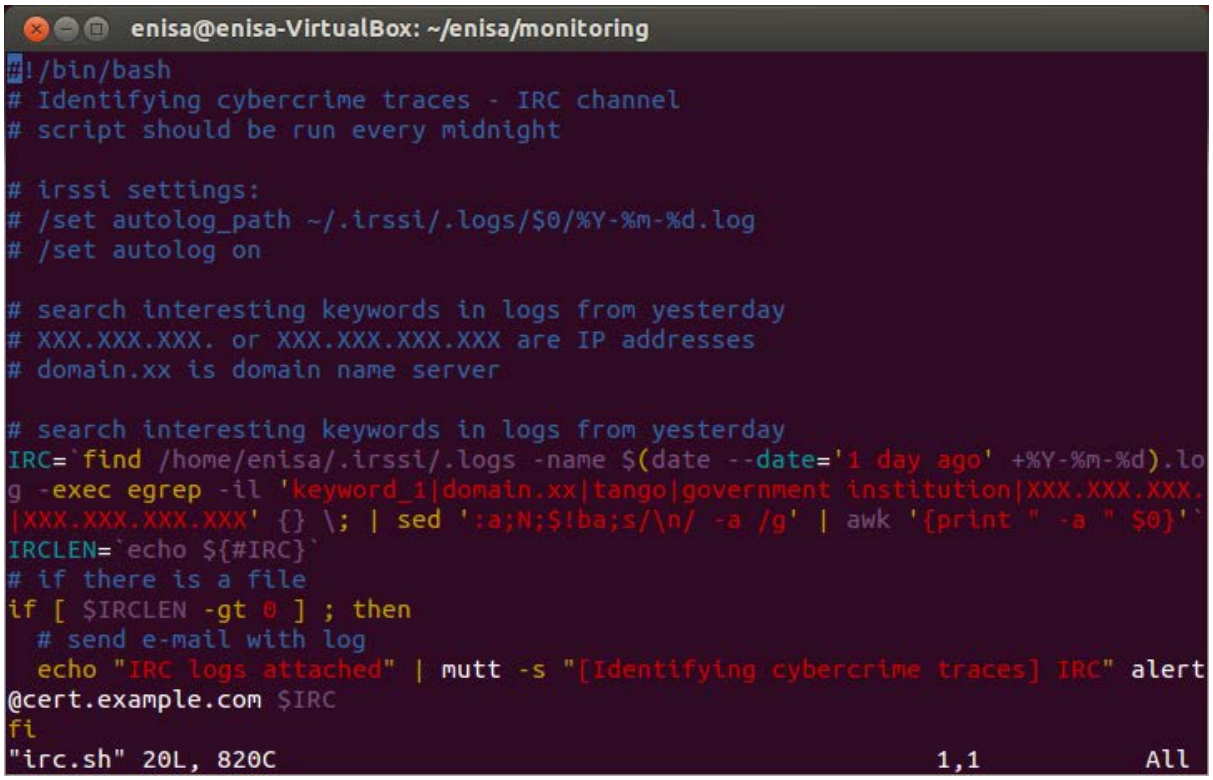

Figure 8: Screenshot from the script for IRC monitoring: script code

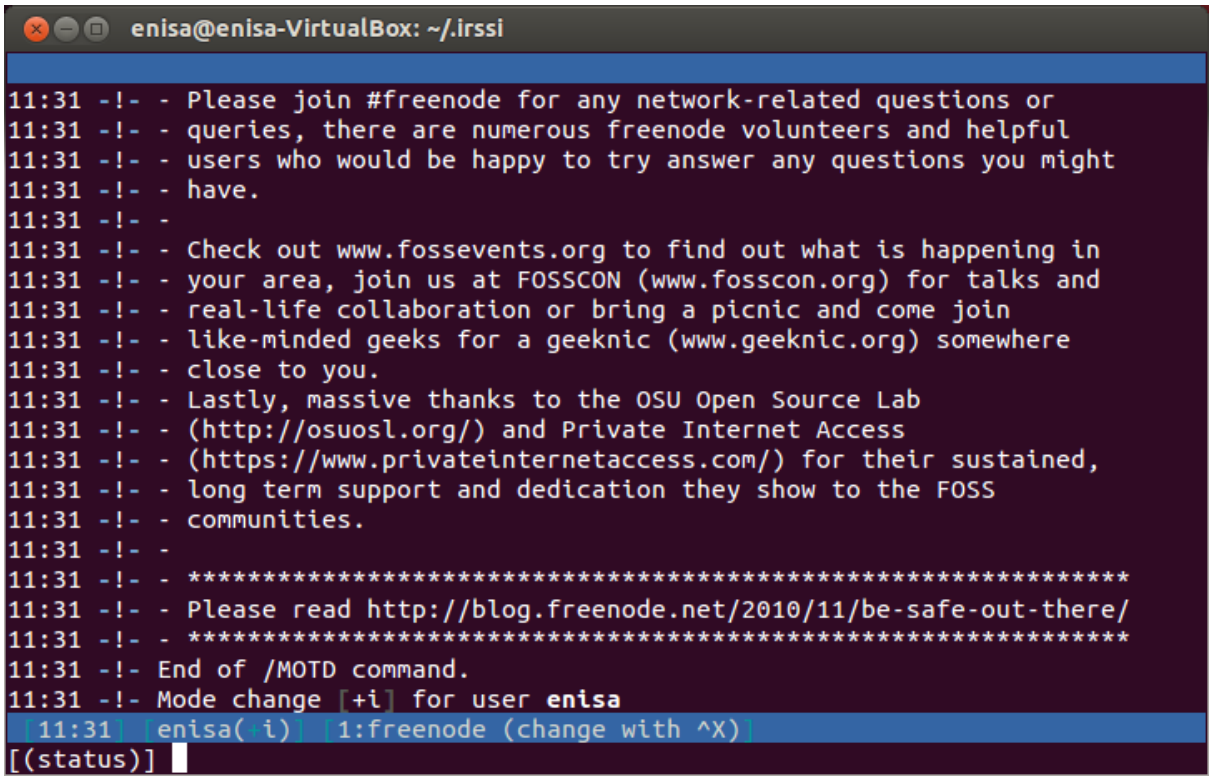

Figure 9: Screenshot from the IRC client IRSSI for IRC monitoring

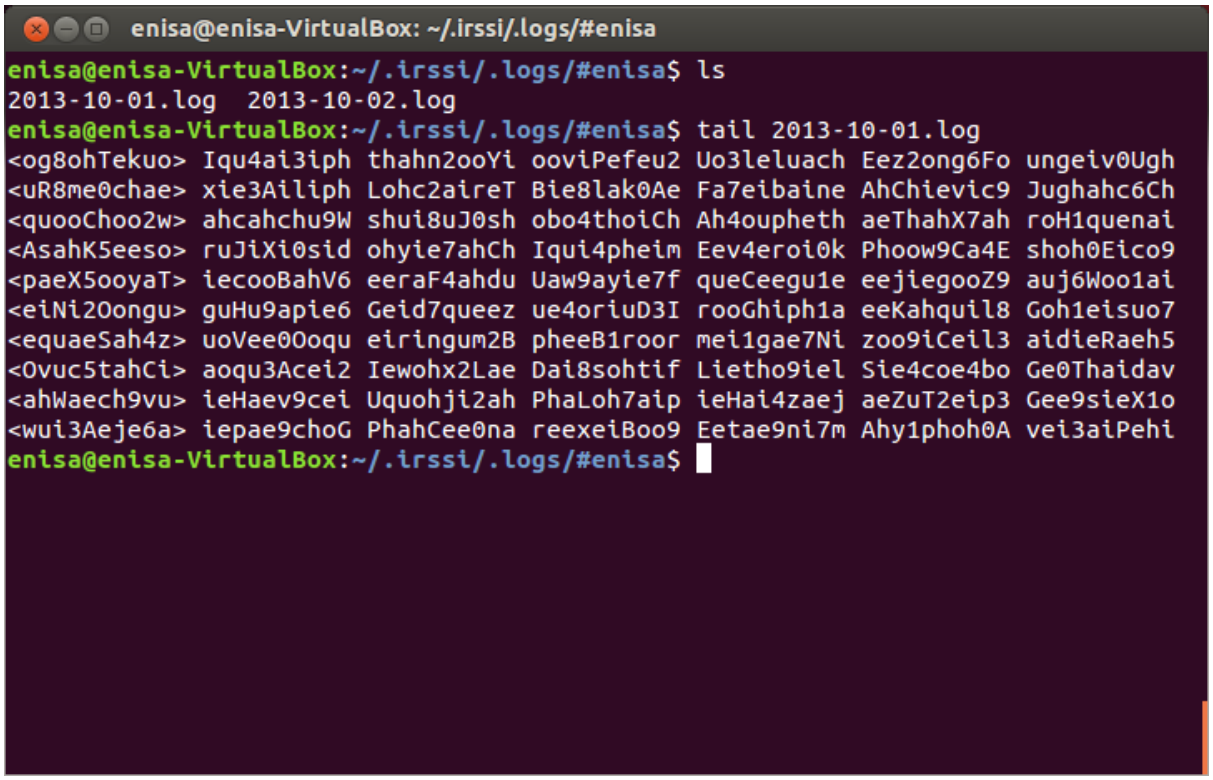

Figure 10: Screenshot of the result of the IRC monitoring

5 Task 3 – Multiple online sources for finding relevant information

This task is to work with various online services that can be used by CERT staff in the process of finding relevant information about particular cases or for constant monitoring of their constituency. During this task the trainees still work in their groups. At the beginning they are asked to work for 20 minutes and list all services in three categories:

Category A – services which they use regularly in their CERT work

Category B – services which they do not use regularly in their CERT work, but they know would be helpful in a particular situation or are worth considering as a regular service in the future

Category C – services understood to have functionality which could be very helpful for CERT staff.

The purpose of this work is to exchange information between trainees in their groups as preparation for the next phase of this task. In the next phase they will go through a number of particular sub-tasks.

5.1 Sub-task 1 – Find all information related to ENISA on Twitter and published during the last week

For this task the participants can use the topsy.com service. They should put 'ENISA' keyword in the search window and choose a specific time range.[7]

[7] *'ENISA' keyword is just an example. During the real exercises trainees can use other specific words more related to the hacking activities or their own constituency representatives. The example can be also an important word related to the current situation (e.g. one of the Anonymous operations or specific hacking group).*

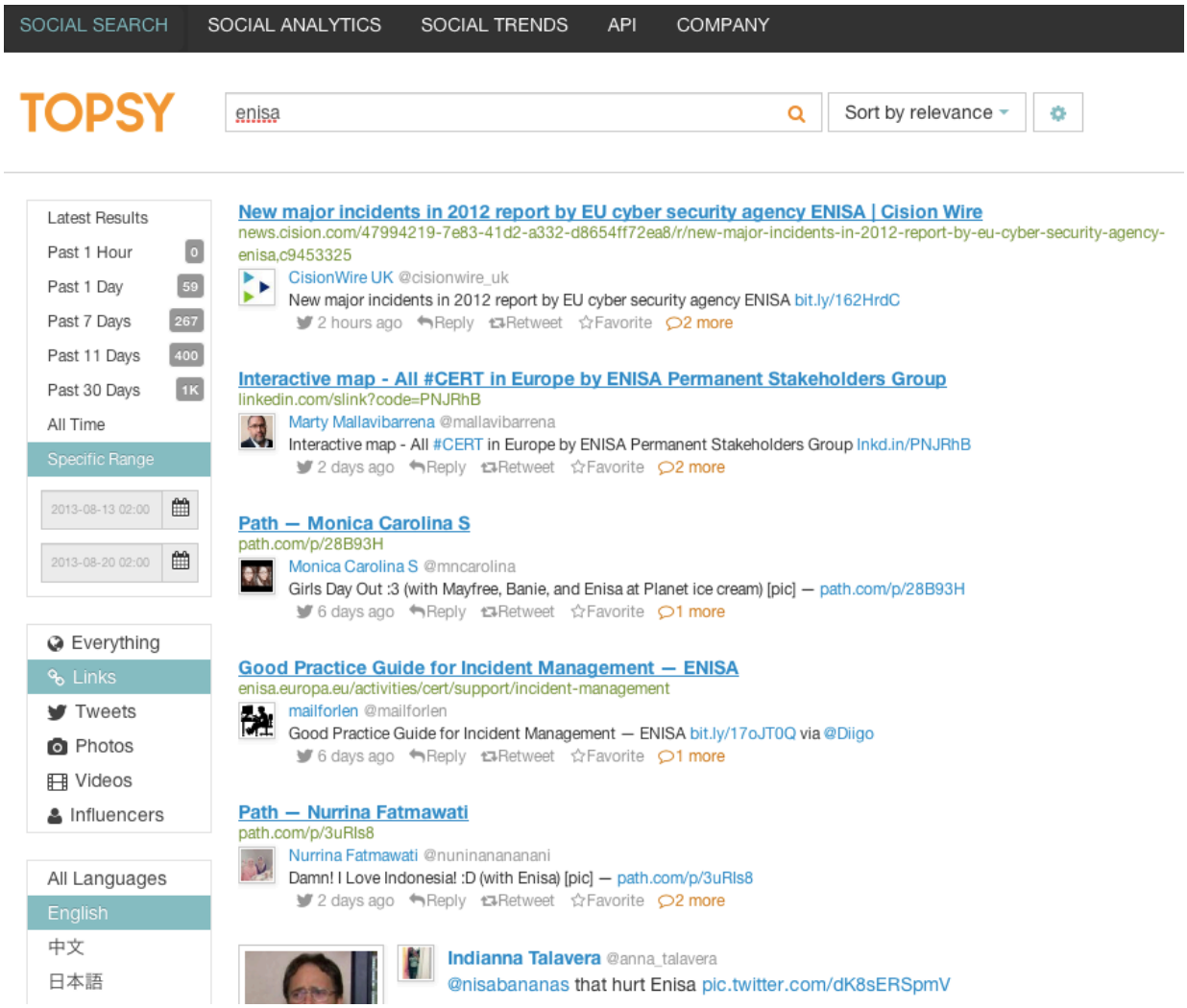

Figure 11: The result of 'ENISA' search in the topsy.com service

Such a general search can come up with a lot of irrelevant information. Thus the trainees are asked to tune the search by excluding false negative results as much as possible, e.g. these which include words like 'love', 'girls', 'Bukvic', 'Custovic'. For this purpose they can use the '-'operator. Then the search query should look like:

Figure 12: Command line for optional usage for topsy.com service

The specific dates should be chosen from the side bar menu:

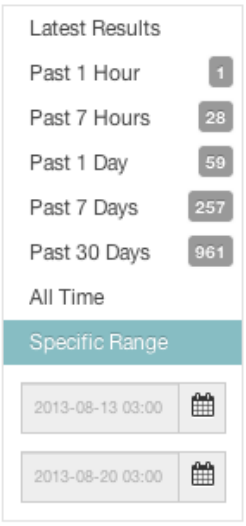

Figure 13: Screenshot from the 'ENISA' searching result in the topsy.com service

5.2 Sub-task 2 – Find all social media services which contain information about a specific user (recognised by a nickname)

Such functionality could be helpful if information about a specific Internet user is important. For this purpose the participant can use the NameChk.com service.

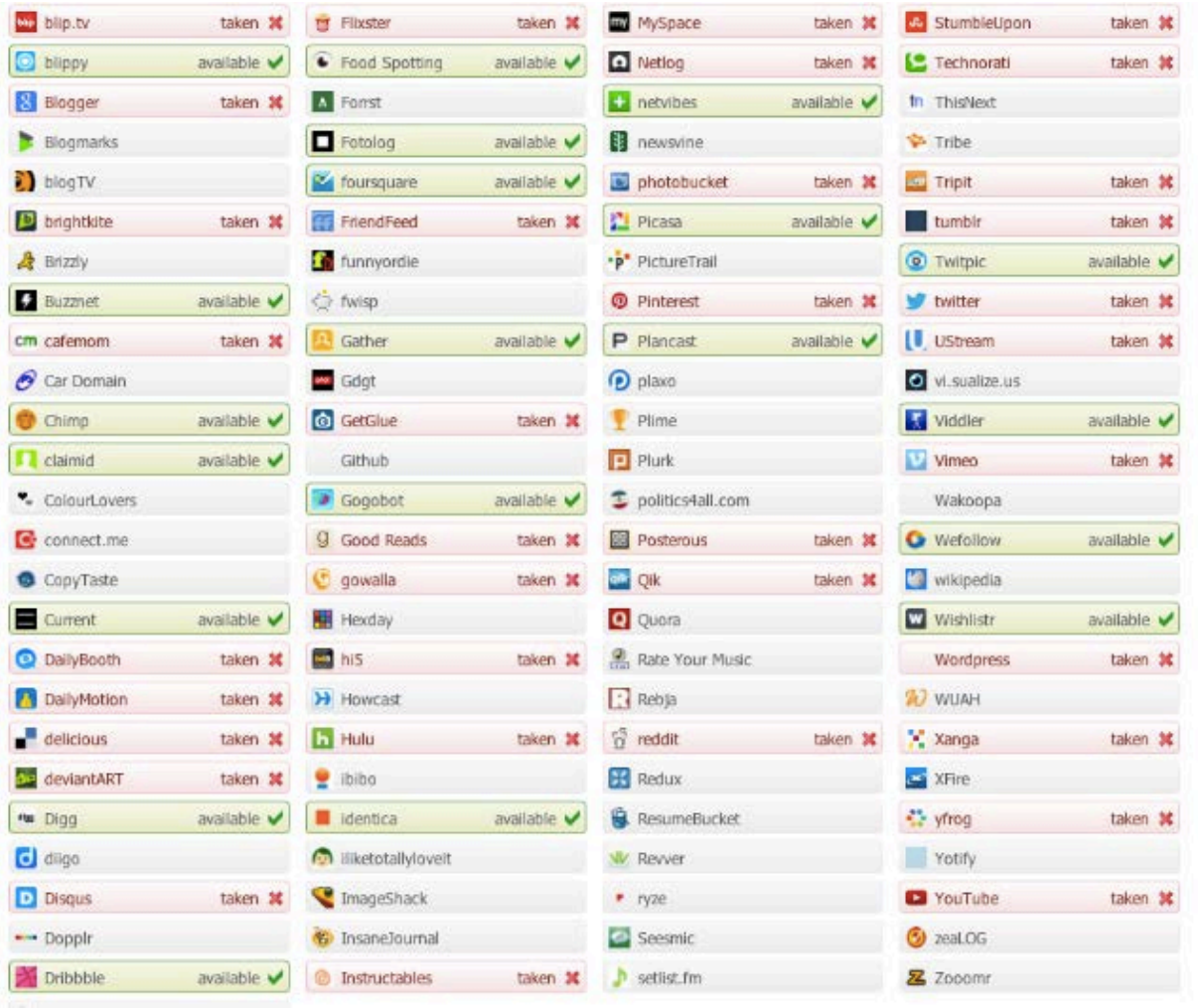

Figure 14: The result of the 'ENISA' search in the namechk.com service

The trainees should undertake an additional task to provide the text list of such accounts which could be used for further monitoring. For this purpose the 'export' function can be used. The result will be as presented below:

5.3 Sub-task 3 – Finding phone numbers and people using PGP software in a particular organisation

If you want to investigate a particular organisation you can probably collect many pieces of information about it. One of them could be phone numbers and PGP keys used in the organisation. The tool which can be used for this purpose is Maltego. Maltego is an open source intelligence and forensics application. It offers mining and gathering of information as well as the representation of this information in an easy-to-understand format. It is available as a free tool for non-commercial purposes.

You can download the tool from the Paterva website at: *http://www.paterva.com/web6/products/download.php*.

The tool installation is intuitive and it should not take more than dozen or so minutes to install. After the installation you will need to register at the website to have access to the public servers used for further investigation.

After the installation the software interface shows the available functionality.

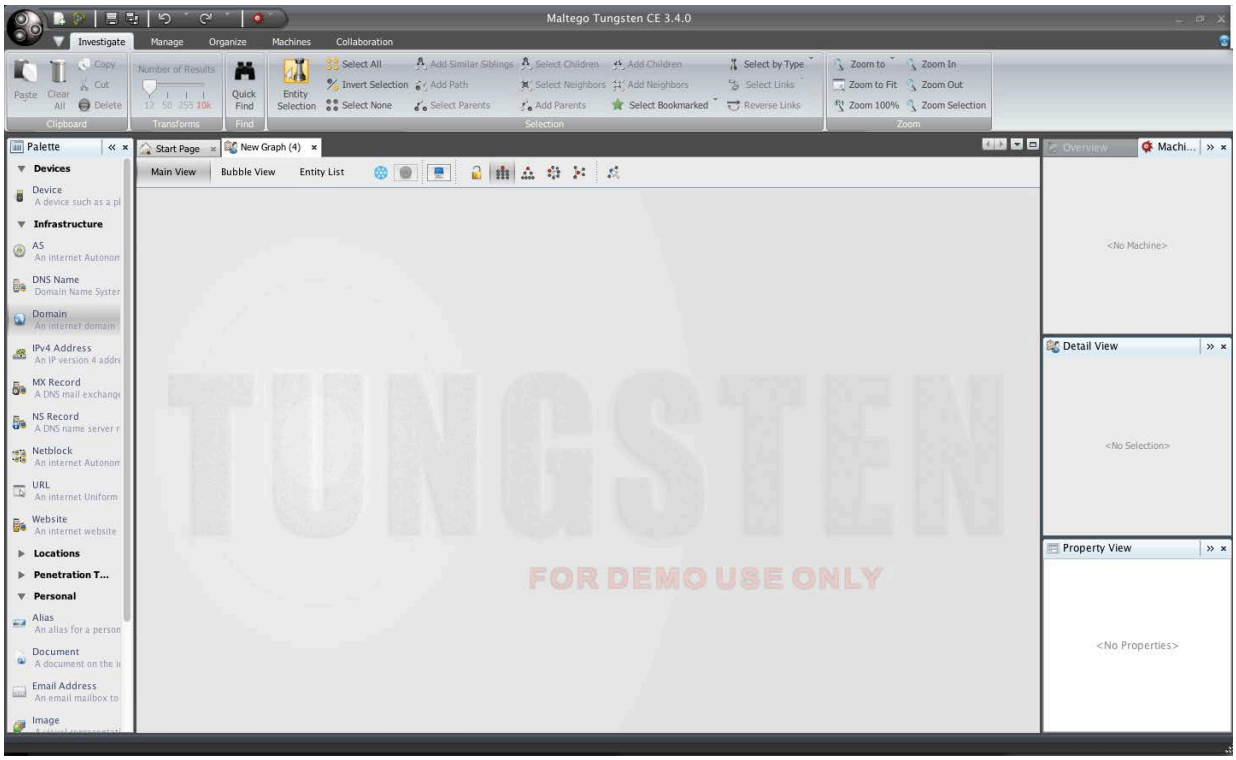

Figure 15: Maltego application

5.3.1 Finding PGP keys

In this sub-task the organisation which will be investigated is ENISA. To select the organisation, the trainees need to choose the domain. This can be done by dragging and dropping the domain palette from the left side bar.

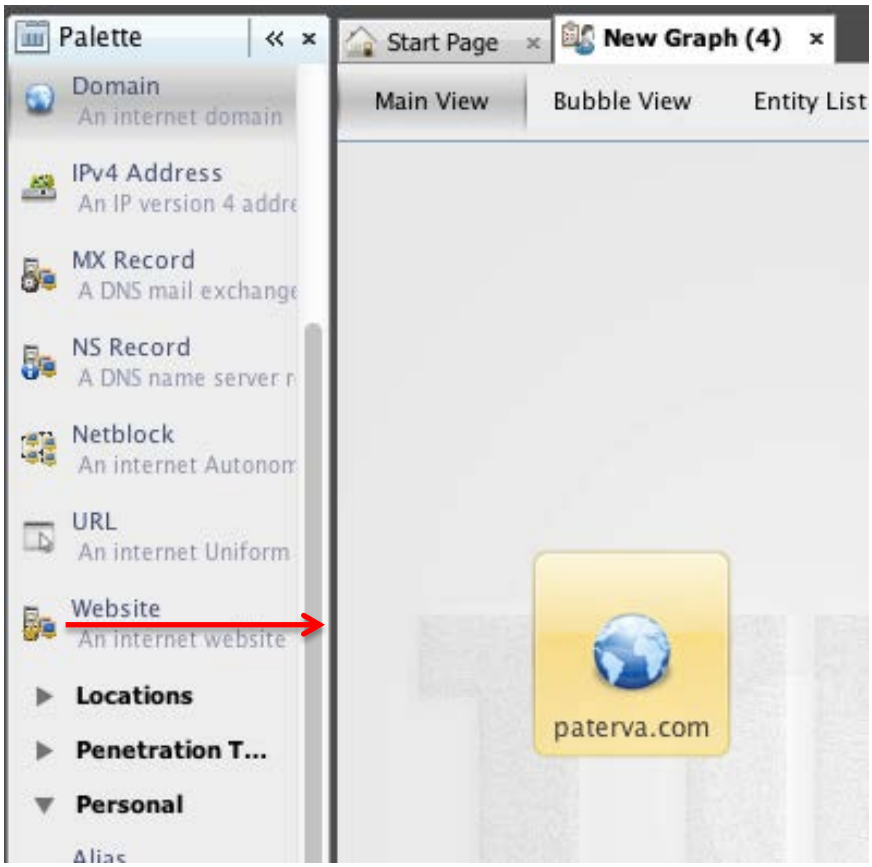

Figure 16: Choosing the organisation by website selection

Having the icon on the main board, by double clicking the name of the organisation can be changed to enisa.europa.eu.

Then the final action can be performed. To receive information about particular PGP keys available in the organisation, the participants should right button click and choose: *Run Transform -> All Transforms -> To Email addresses [PGP].*

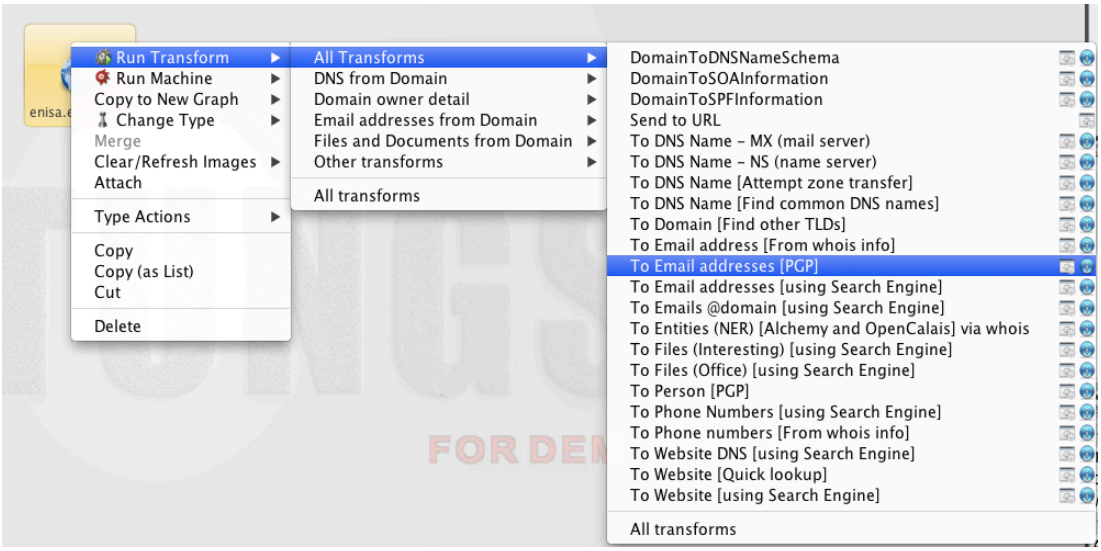

Figure 17: Choosing PGP keys information from the icon menu

After a while the information about the keys will be provided together with the graphical representation.

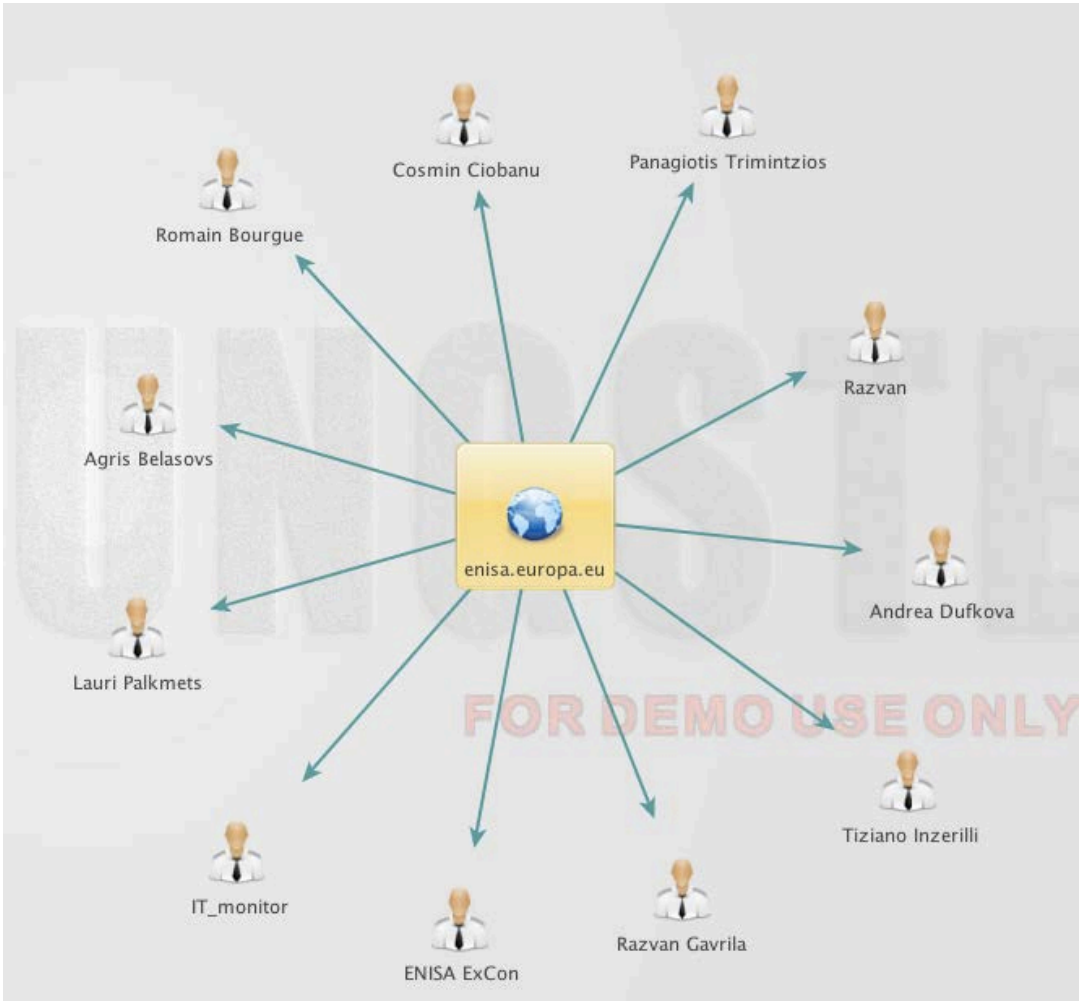

Figure 18: Information about the PGP keys available in the organisation

5.3.2 Finding phone numbers

To add information about the available phone numbers the trainees just need to choose this functionality from the icon menu (right button clicking).

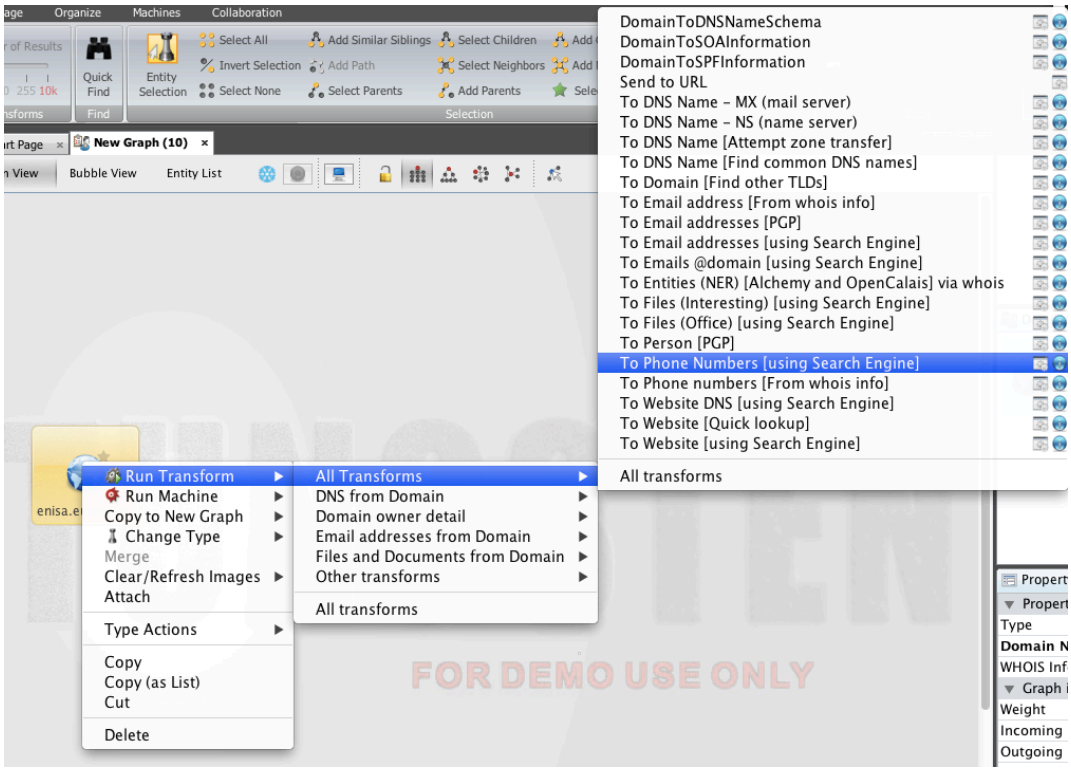

Figure 19: Choosing phone number information from the icon menu

Finally the information about the phone numbers which can be found on the Internet is presented:

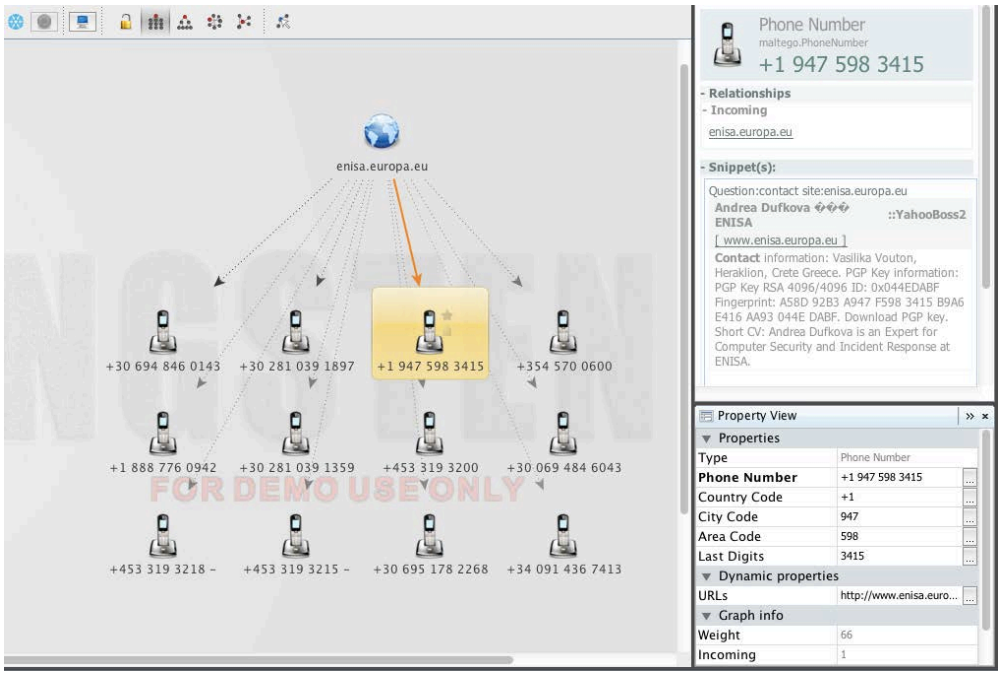

Figure 20: Information about the organisation phone numbers with the property view of one of the numbers

The sub-tasks of Task 3 showed just a few examples of possible tracking activities. At the end of this task the trainees could be asked to discuss other known sources of valuable information. It is important to stress again that they are permitted to use legal methods of information collection only.

Regarding the Maltego Tool, it is worth mentioning that it is a very powerful tool and only a small part of its capabilities was presented during the exercise. It is advised that trainees experiment with this tool and find other possible methods of its usage. If there is enough time they can make a start on this during the exercise. For this purpose the video tutorials prepared by Paterva can be used. They are available at: *http://www.paterva.com/web6/documentation/index.php*.

6 Task 4 – Legal aspects of Internet monitoring services

The task for trainees is to develop a kind of internal legal guide for doing investigation work for their team. They should discuss potential risks in particular.

The potential output of their work should include:

- Establishing relations with law enforcement agencies and:
  - informing them about their planned and ongoing activities;
  - learning about the agencies' needs to modify their monitoring plans;
  - consulting the agencies on whether their activity is legal.
- Planning for the protection of collected information, especially by:
  - ensuring the confidentiality of relevant information;
  - removing irrelevant information;
  - ensuring the integrity of relevant information;
  - developing a simple policy of information distribution.
- Planning for protection of personal data if such will appear as a result of their monitoring.

Then the trainer should ask participants to analyse the laws of particular countries represented by them. Depending on the level of the knowledge among trainees they can still work in their group or they can work all together mentored by the trainer. The result of this exercise should be a matrix; see example below.[8]

Country/Law	Personal Data Protection	Classified Information Protection	Data Breach Notification [etc.]
Poland	Personal data can be processed only if: - there is approval of data subject. - it is necessary to fulfil legal requirements - it is necessary for the public benefit - it is necessary for fulfilling legitimate tasks, which do not violate the data subject's rights	If processed information is classified, then it must: - be revealed only to authorised persons; - be processed in the protected environment (technically and physically) - be protected according to the specified rules described in the special documents which define protection level	ISP must report to data subject about personal data breach in its network not later than 3 days after its discovery. ISP must report to National Data Authority about personal data breach in its network not later than 3 days after discovering it.
Greece			
The Netherlands			
Germany			
[…]			

The trainees should work together in groups and finish the task with the presentation of their findings. This should be followed by a general discussion involving all the trainees, moderated by the trainer. It is important to mention differences in the legal systems of different countries and particular mandates of particular teams (e.g. governmental, private, academic, military, etc.) for performing the content-related network monitoring in their constituencies.

[8] *The proposals in the table are not all examples from the law*.

Special attention should be paid to protection of personal data. Additionally, if the CERT staff want to make public use of collected information, they should be completely sure about its credibility.

7 Summary of the exercise

Use the points below for wrap-up and conclusions during the summary.

- Constant Internet monitoring can be an effective and helpful method of identifying and tracking incident traces.
- There are many tools available for online Internet monitoring.
- A lot of information can be found in the public services.
- Development of your own tools can be a relatively easy and effective method. This process can be fully adjusted to specific needs.
- Language aspects are very important and should be considered carefully when developing own tools for a local constituency.
- Identification, tracking and other information processing can be performed only with full adherence to legal rules, especially those related to personal data protection.

8 Appendix 1 – The code example 1 for network monitoring (Twitter)

To use the script below you have to have a Twitter account and register your application at: https://dev.twitter.com/apps/new

To register your application you can use your Twitter account.

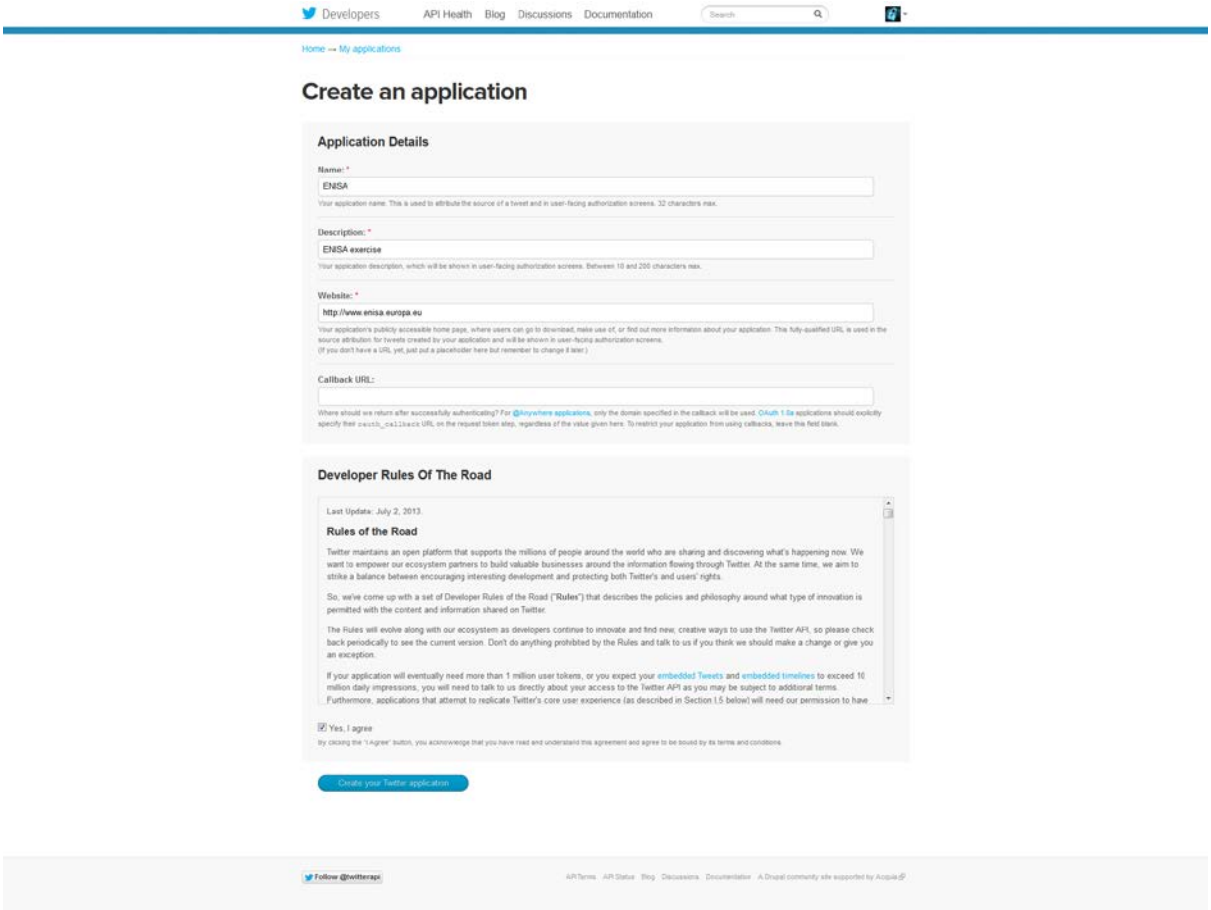

Figure 21: Screenshot presenting the template for creating an application on the Twitter website

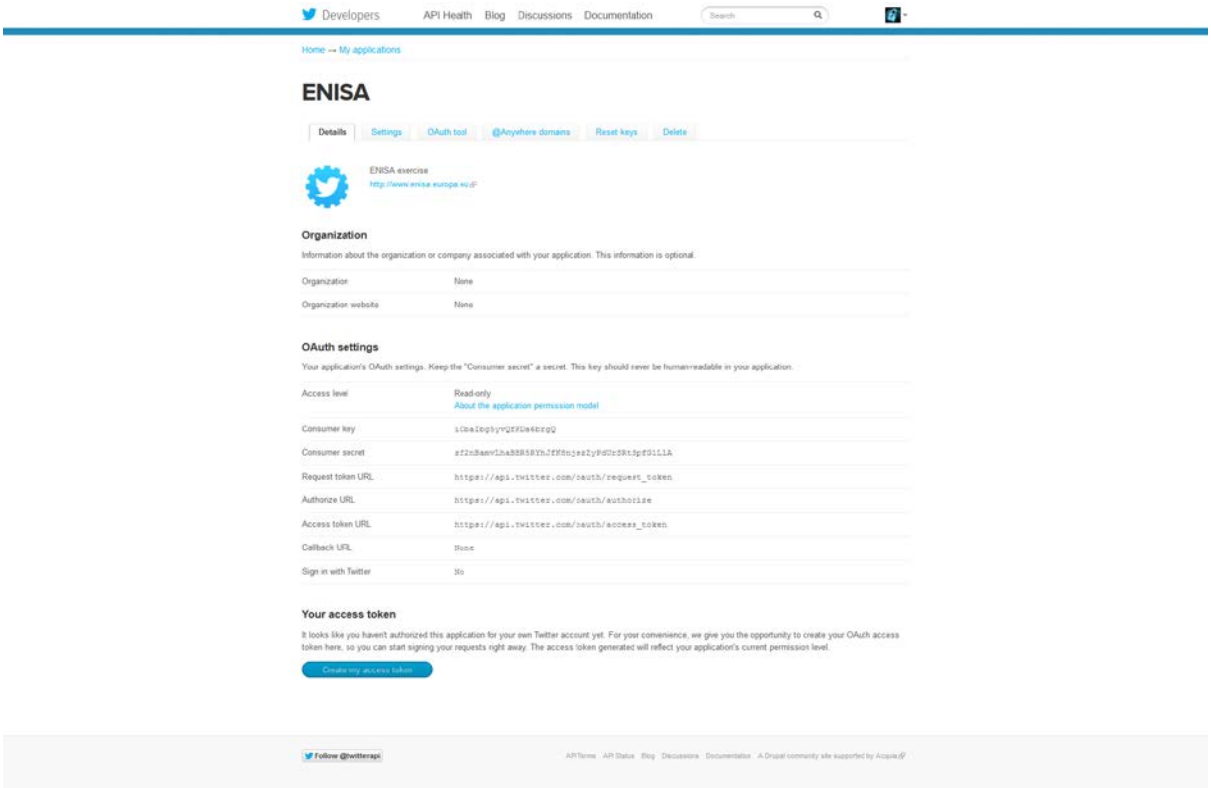

Figure 22: Screenshot presenting information about the registered organisation

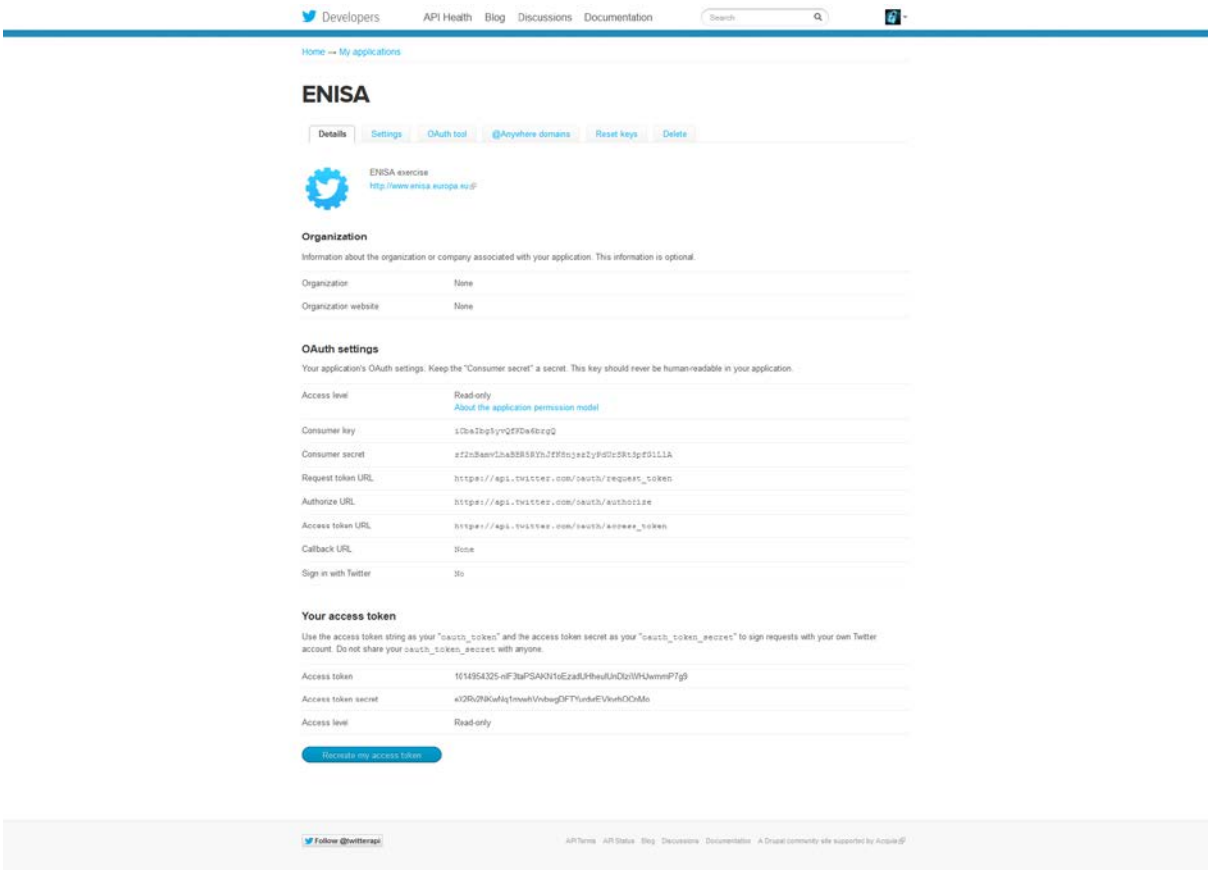

Figure 23: Screenshot presenting the information about the issued token for the application

```
// number of last checked statuses
$notweets = 10;
session_start();
// https://github.com/abraham/twitteroauth
require_once("twitteroauth/twitteroauth/twitteroauth.php");
// Twitter API
$consumerkey = "XXXXX";
$consumersecret = "XXXXX";
$accesstoken = "XXXXX";
$accesstokensecret = "XXXXX";
function getConnectionWithAccessToken($cons_key, $cons_secret, $oauth_token,
$oauth_token_secret) {
   $connection = new TwitterOAuth($cons_key, $cons_secret, $oauth_token, $oauth_token_secret);
   return $connection;
}
$connection = getConnectionWithAccessToken($consumerkey, $consumersecret, $accesstoken,
$accesstokensecret);
$alertarray = array();
$separator = "|#|";
// for each interesting user
foreach ($twitterusers as $twitteruser) {
   $tweets = $connection-
>get("https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name=" . $twitteruser .
"&count=" . $notweets);
   // error handler for Twitter API
   $err = NULL;
   foreach ($tweets->errors as $error) {
      echo $error->message;
      $err = true;
   }
   // stop if there is error
   if ($err) {
      die("\r\n\r\nStop because of above errors.\r\n");
   }
   // for each tweet from each interesting user
```

```
foreach ($tweets as $tweet) {
    // parse time to own format
    $tweetdate = date('Y-m-d H:i:s', strtotime($tweet->created_at));
    // unique id from Twitter id
    $tweetid = $tweet->id_str;
    // create URL from data
    $tweeturl = "https://twitter.com/" . $tweet->user->screen_name . "/status/" . $tweetid;
    $tweettext = $tweet->text;
    // checking if tweet is retweet
    if (substr($tweettext, 0, 4) == 'RT @') {
        $tweetid = $tweet->id_str;
        // if tweet is retweet then add (.=) URL to $tweeturl var
        $tweeturl .= "\r\nhttps://twitter.com/" . $tweet->retweeted_status->user->screen_name .
"/status/" . $tweetid . " (RT)";
        $tweettext = $tweet->retweeted_status->text;
    }
    // check whether the status is in the database
    try {
        // SQLite
        $db = new PDO('sqlite:underground_twitter.sqlite');
        // if there is no database then create
        $db->exec("CREATE TABLE IF NOT EXISTS underground_twitter (id INTEGER PRIMARY KEY, id_str
TEXT UNIQUE NOT NULL)");
        $query = "SELECT COUNT(*) FROM underground_twitter WHERE id_str = '$tweetid'";
        foreach ($db->query($query) as $row) {
            $count = $row["COUNT(*)"];
            echo "$count - $tweetid\r\n";
        }
    } catch (PDOException $e) {
        print 'Exception : ' . $e->getMessage();
    }
    // if SELECT to database return 0 (record doesn't exist in database)
    if ($count == 0) {
        // INSERT INTO database tweet id
```

```
        try {
            $db->exec("INSERT INTO underground_twitter (id_str) VALUES ('" . $tweetid . "');");
        } catch (PDOException $e) {
            print 'Exception : ' . $e->getMessage();
        }
```

```
<?php
/*
 * Identifying cybercrime traces - social media / Twitter
 * script should be run every few (<5) minutes
 */
// interesting keywords
$warningstrings = array("enisa", "agency", "tango", "national government", ".eu, "p0wned", "hacked");
```

Then the Twitter's users should be pointed out:

```
// interesting users
$twitterusers = array("user_01", "user_02", "user_03", "user_04", "user_05");
```

The next step is to use a publicly available Twitter API. The trainer should inform trainees about the availability of such API (https://github.com/abraham/twitteroauth). This helps to build a database with IDs of tweets.

Finally the script should notify the investigators about the discovered threat.

```
        // for each interesting keyword
        foreach ($warningstrings as $warningstring) {
            // check url
            foreach ($tweet->entities->urls as $turl) {
                if (strpos($turl->expanded_url, $warningstring) !== false) {
                    $alert = $tweetdate . $separator . $tweeturl . $separator . $tweettext . $separator;
                    // add to alert array
                    array_push($alertarray, $alert);
                }
            }
            // check tweet
            if (strpos($tweettext, $warningstring) !== false) {
                $alert = $tweetdate . $separator . $tweeturl . $separator . $tweettext . $separator;
                // add to alert array
```

```
                array_push($alertarray, $alert);
            }
        }
    }
  }
}
$alertuniquearray = array_unique($alertarray);
$alertnotify = "WARNING ALERTS:\r\n";
$alertstrlen = strlen($alertnotify);
foreach ($alertuniquearray as $alert) {
  $alertnotify .= "\r\n" . str_replace($separator, "\r\n", $alert);
}
$alertstrlencheck = strlen($alertnotify);
// if there is at least one new tweet
if ($alertstrlen < $alertstrlencheck) {
  echo $alertnotify;
  // send e-mail
  mail('alert@our-cert.eu', '[Identifying cybercrime traces] Twitter', $alertnotify);
}
?>
```

9 Appendix 2 – The code example for visual presentation of the tweets searching

```
<html>
  <head>
    <script type="text/javascript" src="https://www.google.com/jsapi"></script>
    <script type="text/javascript">
      google.load("visualization", "1", {packages: ["corechart"]});
      google.setOnLoadCallback(drawChart);
      function drawChart() {
        var data = google.visualization.arrayToDataTable([
<?php
// https://github.com/abraham/twitteroauth
require_once("/home/enisa/enisa/monitoring/twitteroauth/twitteroauth/twitteroauth.php");
// Twitter API
$consumerkey = "XXXXX";
$consumersecret = "XXXXX";
$accesstoken = "XXXXX";
$accesstokensecret = "XXXXX";
function getConnectionWithAccessToken($cons_key, $cons_secret, $oauth_token,
$oauth_token_secret) {
  $connection = new TwitterOAuth($cons_key, $cons_secret, $oauth_token, $oauth_token_secret);
  return $connection;
}
$connection = getConnectionWithAccessToken($consumerkey, $consumersecret, $accesstoken,
$accesstokensecret);
// https://dev.twitter.com/docs/api/1.1/get/search/tweets
// https://dev.twitter.com/docs/using-search
$tweets = $connection-
>get("https://api.twitter.com/1.1/search/tweets.json?q=%23enisa%20exclude:retweets&count=100
");
$stack = array();
foreach ($tweets->statuses as $tweet) {
  //var_dump($tweet);
  $tweetdate = date('Y-m-d H:i:s', strtotime($tweet->created_at));
  $tweettext = $tweet->text;
```

```
    //echo $tweetdate . "\n";
    array_push($stack, explode(' ', $tweetdate)[0]);
}
$stack = array_count_values($stack);
$chart = "\t\t\t\t\t['Day', '#ENISA'],\n";
while ($pie = current($stack)) {
    $chart .= "\t\t\t\t\t['" . key($stack) . "', $pie],\n";
    next($stack);
}
$chart = substr($chart, 0, -2);
echo $chart;
?>
            ]);
            var options = {
               title: '#ENISA in last 100 tweets'
            };
            var chart = new google.visualization.ColumnChart(document.getElementById('chart_div'));
            chart.draw(data, options);
          }
      </script>
    </head>
    <body>
      <div id="chart_div" style="width: 900px; height: 500px;"></div>
    </body>
</html>
```

10 Appendix 3 – The code example for IRC monitoring

```
#!/bin/bash
# Identifying cybercrime traces - IRC channel
# script should be run every midnight
# irssi (as IRC client) settings:
# /set autolog_path ~/.irssi/.logs/$0/%Y-%m-%d.log
# /set autolog on
# search interesting keywords in logs from yesterday
# XXX.XXX.XXX. or XXX.XXX.XXX.XXX are IP addresses
# domain.xx is domain name server
IRC=`find /home/XXXXX/.irssi/.logs -name $(date --date='1 day ago' +%Y-%m-%d).log -exec egrep -il 'keyword_1|domain.xx|tango|government institution|XXX.XXX.XXX.|XXX.XXX.XXX.XXX' {} \; | sed ':a;N;$!ba;s/\n/ -a /g' | awk '{print " -a " $0}'`
IRCLEN=`echo ${#IRC}`
```

If the keyword is found then the investigators should be notified via email.

```
# if there is a file
if [ $IRCLEN -gt 0 ] ; then
  # send e-mail with log
  echo "IRC logs attached" | mutt -s "[Identifying cybercrime traces] IRC" alert@our-cert.eu $IRC
Fi
```

The result of the script is a mail which contains the matched word. In the example, due to the content monitoring of the channel “hackchat” the keyword “org_one.gov.eu” was discovered in the conversation of criminals. Logs are:

--- Log opened Sun Jun 29 00:00:13 2013

00:04 -!- hacker01_ [hacker01^anonmx@AN-di2.t5r4.govi.IP] has joined #hackchat

00:04 -!- hacker01_ is "hacker01 AnonSomwhere" on (unknown)

00:07 -!- hacker01__ [hacker01^anonmx@AN-di2.t5r4.govi.IP] has joined #hackchat

00:07 -!- hacker01_ [hacker01^anonmx@AN-di2.t5r4.govi.IP] has quit [Connection closed]

00:07 -!- hacker01__ is "hacker01 AnonSomwhere" on (unknown)

00:08 -!- hacker01 [hacker01^anonmx@AN-di2.t5r4.govi.IP] has quit [Pingtimeout: 121 seconds]

00:09 -!- hacker01__ [hacker01^anonmx@AN-di2.t5r4.govi.IP] has quit [Connection closed]

00:09 -!- Odik__ [Ella__@AN-20e.6tt.42adcl.IP] has joined #hackchat

00:09 -!- Odik__ is "Bluto" on (unknown)

00:09 -!- Dragon [Dragon@AN-j3b.im0.23nk0v.IP] has quit [Connection closed]

00:09 -!- wawka [not@yourhouse.anymore] has quit [Connection closed]

00:09 -!- wawka [not@yourhouse.anymore] has joined #hackchat

00:09 -!- wawka is "..." on (unknown)

00:09 -!- Dragon_ [Dragon@AN-j3b.im0.23nk0v.IP] has joined #hackchat

00:09 -!- Dragon_ is "Dragon" on (unknown)

00:10 -!- hacker01 [hacker01^anonmx@AN-di2.t5r4.govi.IP] has joined #hackchat

00:10 -!- hacker01 is "hacker01 AnonSomwhere" on (unknown)

00:12 -!- Ella_ [Ella__@AN-20e.6tt.42adcl.IP] has quit [Ping timeout: 121 seconds]

00:17 -!- hacker01 [hacker01^anonmx@AN-di2.t5r4.govi.IP] has quit [Connection closed]

00:18 -!- hacker01 [hacker01^anonmx@AN-di2.t5r4.govi.IP] has joined #hackchat

00:18 -!- hacker01 is "hacker01 AnonSomwhere" on (unknown)

00:20 -!- Dragon is "Dragon" on (unknown)

00:20 -!- Dragon [Dragon@AN-j3b.im0.23nk0v.IP] has joined #hackchat

00:20 < Dragon> hi, let's hack org_one.gov.eu

00:20 -!- Ella_ [Ella__@AN-20e.6tt.42adcl.IP] has joined #hackchat

00:20 -!- Ella_ is "Bluto" on (unknown)

00:21 < hacker01> ok, let's do it

01:21 -!- Dragon_ [Dragon@AN-j3b.im0.23nk0v.IP] has quit [Ping timeout: 121 seconds]

01:22 -!- hacker01 [hacker01^anonmx@AN-di2.t5r4.govi.IP] has quit [Connection closed]

01:22 -!- L3JI0N [LEJION@1671671] has quit [Ping timeout: 121 seconds]

--- Log closed Jun 29 01:22:25 2013

11 References

1. 'Hacking duo charged with DDoSing Amazon, then bragging about it', *http://arstechnica.com/security/2012/07/hacking-duo-charged-for-amazon-ddos/*
2. 'Hackers Claim to Have PlayStation Users' Card Data', *http://bits.blogs.nytimes.com/2011/04/28/hackers-claim-to-have-playstation-users-card-data/?_r=0*
3. The Personal Data Protection Law in the EU Member States based on the same directive – 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications). Its content (in official EU languages) is available at: *http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0058:EN:NOT*

ENISA
European Union Agency for Network and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office
1 Vass. Sofias & Meg. Alexandrou
Marousi 151 24, Athens, Greece

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu