



# Identifying and handling cybercrime traces

*Toolset, Document for students*

September 2013





## 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](http://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 Weźgowiec 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](mailto:CERT-Relations@enisa.europa.eu)

For media enquiries about this document, please use [press@enisa.europa.eu](mailto: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**

Notice must be taken that this publication represents the views and interpretations of the authors and editors, unless stated otherwise. This publication should not be construed to be a legal action of ENISA or the ENISA bodies unless adopted pursuant to the Regulation (EU) No 526/2013. This publication does not necessarily represent state-of-the-art and ENISA may update it from time to time.

Third-party sources are quoted as appropriate. ENISA is not responsible for the content of the external sources including external websites referenced in this publication.

This publication is intended for information purposes only. It must be accessible free of charge. Neither ENISA nor any person acting on its behalf is responsible for the use that might be made of the information contained in this publication.

**Copyright Notice**

© European Union Agency for Network and Information Security (ENISA), 2013

Reproduction is authorised provided the source is acknowledged.

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



## Table of Contents

<b>1</b>	<b>General Description</b>	<b>1</b>
<b>2</b>	<b>Introduction</b>	<b>3</b>
<b>3</b>	<b>Task 1 – Social Media (based on Twitter example)</b>	<b>4</b>
3.1	Subtask 1 – Twitter accounts.....	4
3.2	Subtask 2 – Keywords .....	4
3.3	Subtask 3 – Code development .....	5
3.4	Conclusion of Task 1.....	8
<b>4</b>	<b>Task 2 – IRC channels</b>	<b>9</b>
<b>5</b>	<b>Task 3 – Multiple online sources for finding relevant information</b>	<b>12</b>
5.1	Sub-task 1 – Find all information related to ENISA on Twitter and published during the last week...	12
5.2	Sub-task 2 – Find all social media services which contain information about a specific user (recognised by a nickname) .....	13
5.3	Sub-task 3 – Finding phone numbers and people using PGP software in a particular organisation...	13
<b>6</b>	<b>Task 4 – Legal aspects of Internet monitoring services</b>	<b>18</b>
<b>7</b>	<b>Appendix 1 – The code example 1 for network monitoring (Twitter)</b>	<b>19</b>
<b>8</b>	<b>Appendix 2 – The code example for visual presentation of the tweets searching</b>	<b>25</b>
<b>9</b>	<b>Appendix 3 – The code example for IRC monitoring</b>	<b>27</b>
<b>10</b>	<b>References</b>	<b>29</b>

## 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.<sup>1</sup>
- Hackers discussed break-in activity into the Sony PlayStation Network and the fact of credit card numbers possession on an underground Internet forum.

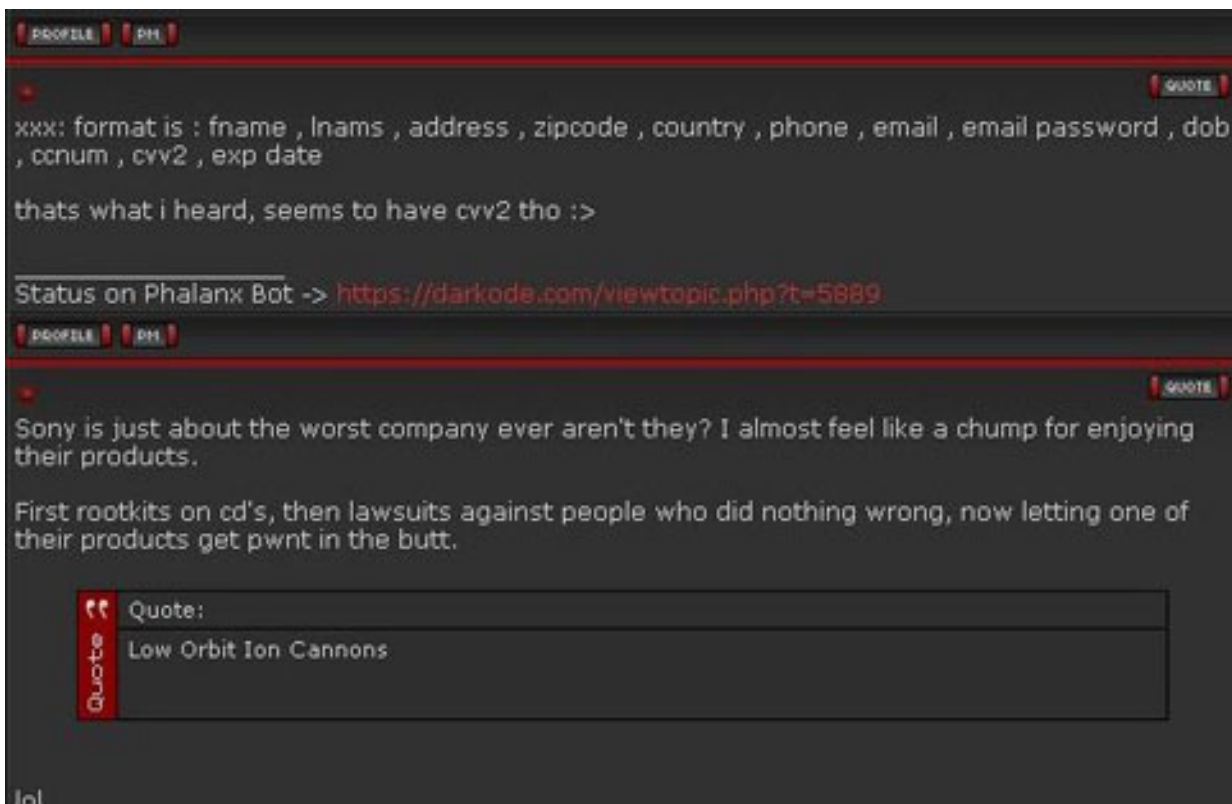


Figure 1: Hacker forum screenshot presenting discussion about the types of data hackers stole from Sony<sup>2</sup>

In the first two subtasks of Task 1 (*Twitter accounts* and *keywords*) you will learn about some rules for content-related network monitoring and you will use this knowledge in Task 2 (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:

<sup>1</sup> More: <http://arstechnica.com/security/2012/07/hacking-duo-charged-for-amazon-ddos/>

<sup>2</sup> Source: New York Times online service: [http://bits.blogs.nytimes.com/2011/04/28/hackers-claim-to-have-playstation-users-card-data/?\\_r=0](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, you will learn about legal aspects of Internet monitoring activities. It is obvious that you and other trainees come from different countries with different legal systems, but some general rules related to this topic can be learnt. The example presented in this exercise will be based upon national legislation, the most important aspect of which relates to Personal Data Protection Law.<sup>3</sup>

---

<sup>3</sup> 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

At the beginning of the exercise the trainer will inform you about the following key factors:

- The CERT is responsible for a specific constituency;
- part of the CERT services is network monitoring for active detection.<sup>4</sup>

Your area of interest should be various information sources, where you can detect a potential incident, not only from the technical point of view, but also from the point of view of identifying the attackers. You should also take into consideration all legal aspects related to the performance of your work according to existing law in your country. Special attention should be paid to activities that 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, data anonymisation, attribution and handing over the assumptions made. It is also important that use of publicly available systems for content monitoring be kept to a minimum. The main reason is to avoid the 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<sup>5</sup>

There are two main sources of information, 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 on a computer.

<sup>4</sup> Classic IP threats monitoring is not the main task during this exercise

<sup>5</sup> 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.

From now on you will be assigned to one of the groups.

#### 3.1 Subtask 1 – Twitter accounts

The *first subtask* for you is to discuss and determine Twitter user accounts which, in your opinion, could become significant sources of relevant information. You can use your favourite Twitter accounts as well as carrying out Internet research during the exercise to collect new favourites. The guidance for selection are:

- your 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'<sup>6</sup>;
- their geographical location does matter, e.g. 'AnonInPoland' user channel;
- some periodic actions/operations can bring relevant information, e.g. '#OpUSA';
- you 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 your group should present your proposals of Twitter accounts. Prepare a short presentation (in the form of a list of keywords). Also suggest hackers' slang terms which you can use in your investigations.

#### 3.2 Subtask 2 – Keywords

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 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);
- 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';

---

<sup>6</sup> Sample terms



- words usually used when information about successful attacks is issued, e.g.: ‘tango down’, ‘p0wned’, ‘hacked’. If local language words are also often used in such situation, they should be added to the set.

Now you should present your proposals of keywords. This is a short presentation (in the form of a simple list). Think about the quality and usefulness of the proposals – yours and these presented by others.

### 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:

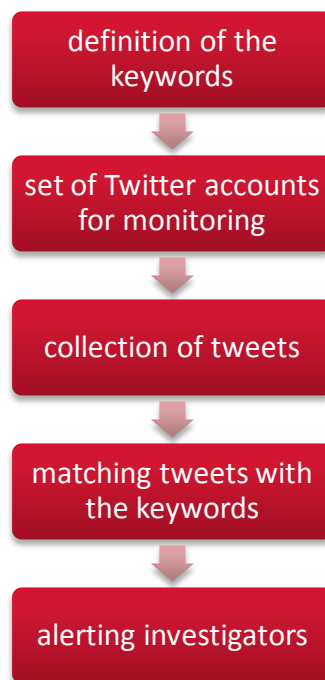


Figure 3: Modules of the Twitter monitoring code

For the code development purposes you can use the common set examples.

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.

```
enisa@enisa-VirtualBox: ~/enisa/monitoring
?php
/*
 * Identifying cybercrime traces - Twitter
 * script should be run every few (<5) minutes
 */

// interesting keywords
$warningstrings = array("tango down", "hacked");
// interesting users
$twitterusers = array("m3gaz0rd");

// number of last checked statuses
$notweets = 10;

session_start();

// https://github.com/abraham/twitteroauth
// git clone git://github.com/abraham/twitteroauth.git
require_once("twitteroauth/twitteroauth/twitteroauth.php");

// Twitter API
// https://dev.twitter.com/apps/new
$consumerkey = "XXXXX";
"twitter.php" 134L, 4722C 1,1 Top
```

Figure 4: Screenshot of the script for Twitter monitoring

```
enisa@enisa-VirtualBox: ~/enisa/monitoring
enisa@enisa-VirtualBox:~/enisa/monitoring$ sqlite3 twitter.sqlite
SQLite version 3.7.15.2 2013-01-09 11:53:05
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .tables
twitter
sqlite> select * from twitter limit 10;
227|116441232158363648
226|116443573431451648
225|118276500167462913
224|118797825550200832
223|141110393161527296
222|141110482613436416
221|141116007698808832
100|154831560204496896
99|154883856032473089
98|155349302804357120
sqlite>
```

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

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

In this example keyword 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 II #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:

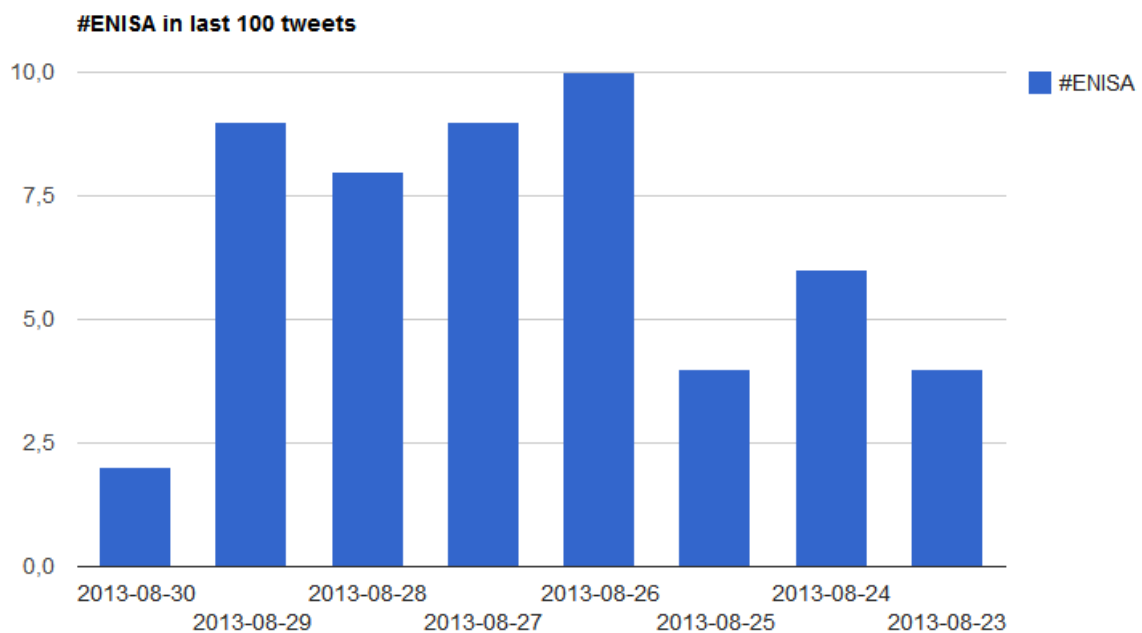


Figure 6: The graphical representation of the ‘ENISA’ keyword

```
enisa@enisa-VirtualBox: ~
require_once("/home/enisa/enisa/monitoring/twitteroauth/twitteroauth/twitteraut
h.php");

// Twitter API
$consumerkey = "XXXXX";
$consumersecret = "XXXXX";
$accesstoken = "XXXXX";
$accesstokensecret = "XXXXX";

function getConnectionWithAccessToken($cons_key, $cons_secret, $oauth_token, $oa
uth_token_secret) {
    $connection = new TwitterOAuth($cons_key, $cons_secret, $oauth_token, $oauth
_token_secret);
    return $connection;
}

$connection = getConnectionWithAccessToken($consumerkey, $consumersecret, $acces
stoken, $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=%23
enisa%20exclude:retweets&count=100");
```

28,1 21%

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

### 3.4 Conclusion of Task 1

After the task has been completed your trainer will lead a wrap-up session. During the session you will discuss:

- Effectiveness of the search of Twitter channel;
- Legal considerations related to this kind of search;
- Your experiences and ideas for effective monitoring of social media;
- The most interesting keywords (including hackers' slang words) for effective search;
- Examples of how the graphical representation for Twitter monitoring solution worked.

## 4 Task 2 – IRC channels

During this task you will improve your 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 identification of the fact that the presence on a channel is only for monitoring and discovering criminal activity. Such identification can provoke attacks on an investigating party.

You should develop your own ideas on how to challenge the above problems.

The main ideas can be:

- to use anonymisation of the network connectivity (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 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 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.

```

enisa@enisa-VirtualBox: ~/enisa/monitoring
~/bin/bash
# Identifying cybercrime traces - IRC channel
# script should be run every midnight

# irssi 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

# search interesting keywords in logs from yesterday
IRC=`find /home/enisa/.irssi/.logs -name $(date --date='1 day ago' +%Y-%m-%d).log
g -exec egrep -il 'keyword_1|domain.xx|tango|government institution|XXX.XXX.XXX.
|XXX.XXX.XXX.XXX' {} \; | sed ':a;N;$!ba;s/\n/ -a /g' | awk '{print " -a " $0}'`
IRCLLEN=`echo ${#IRC}`
# if there is a file
if [ $IRCLLEN -gt 0 ] ; then
  # send e-mail with log
  echo "IRC logs attached" | mutt -s "[Identifying cybercrime traces] IRC" alert
@cert.example.com $IRC
fi
"irc.sh" 20L, 820C                                     1,1      All
  
```

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

```

enisa@enisa-VirtualBox: ~/.irssi
11:31 -!- - Please join #freenode for any network-related questions or
11:31 -!- - queries, there are numerous freenode volunteers and helpful
11:31 -!- - users who would be happy to try answer any questions you might
11:31 -!- - have.
11:31 -!- -
11:31 -!- - Check out www.fossevents.org to find out what is happening in
11:31 -!- - your area, join us at FOSSCON (www.fosscon.org) for talks and
11:31 -!- - real-life collaboration or bring a picnic and come join
11:31 -!- - like-minded geeks for a geeknic (www.geeknic.org) somewhere
11:31 -!- - close to you.
11:31 -!- - Lastly, massive thanks to the OSU Open Source Lab
11:31 -!- - (http://osuosl.org/) and Private Internet Access
11:31 -!- - (https://www.privateinternetaccess.com/) for their sustained,
11:31 -!- - long term support and dedication they show to the FOSS
11:31 -!- - communities.
11:31 -!- -
11:31 -!- - *****
11:31 -!- - Please read http://blog.freenode.net/2010/11/be-safe-out-there/
11:31 -!- - *****
11:31 -!- - End of /MOTD command.
11:31 -!- - Mode change [+i] for user enisa
11:31 enisa( i) 1:freenode (change with ^X)
[(status)]
  
```

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

```
enisa@enisa-VirtualBox: ~/.irssi/logs/#enisa
enisa@enisa-VirtualBox: ~/.irssi/logs/#enisa$ ls
2013-10-01.log 2013-10-02.log
enisa@enisa-VirtualBox: ~/.irssi/logs/#enisa$ tail 2013-10-01.log
<og8ohTekuo> Iqu4ai3iph thahn2ooYi ooviPefeu2 Uo3leluach Eez2ong6Fo ungeiv0Ugh
<uR8me0chae> xie3Ailiph Lohc2aireT Bie8lak0Ae Fa7eibaine AhChievic9 Jughahc6Ch
<quooChoo2w> ahcahchu9W shui8uJ0sh obo4thoiCh Ah4oupheth aeThahX7ah roH1quenai
<AsahK5eeso> ruJiXi0sid ohyie7ahCh Iqui4pheim Eev4eroi0k Phoow9Ca4E shoh0Eico9
<paeX5ooyaT> iecooBahV6 eeraF4ahdu Uaw9ayie7f queCeegu1e eejegooZ9 auj6Woo1ai
<eiNi20ongu> guHu9apie6 Geid7queez ue4oriuD3I rooGhiph1a eeKahquil8 Goh1eisuo7
<equaeSah4z> uoVee00oqu eiringum2B pheeB1roor mei1gae7Ni zoo9iCeil3 aidieRaeh5
<Ovuc5tahCi> aoqu3Acei2 Iewohx2Lae Dai8sohtif Lietho9iel Sie4coe4bo Ge0Thaidav
<ahWaech9vu> ieHaev9cei Uquohji2ah PhaLoh7aip ieHai4zaej aeZuT2eip3 Gee9sieX1o
<wui3Aeje6a> iepae9choG PhahCee0na reexeiBoo9 Eetae9ni7m Ahy1phoh0A vei3aiPehi
enisa@enisa-VirtualBox: ~/.irssi/logs/#enisa$
```

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 you will also work in groups. At the beginning please work for 20 minutes and list all services in three categories:

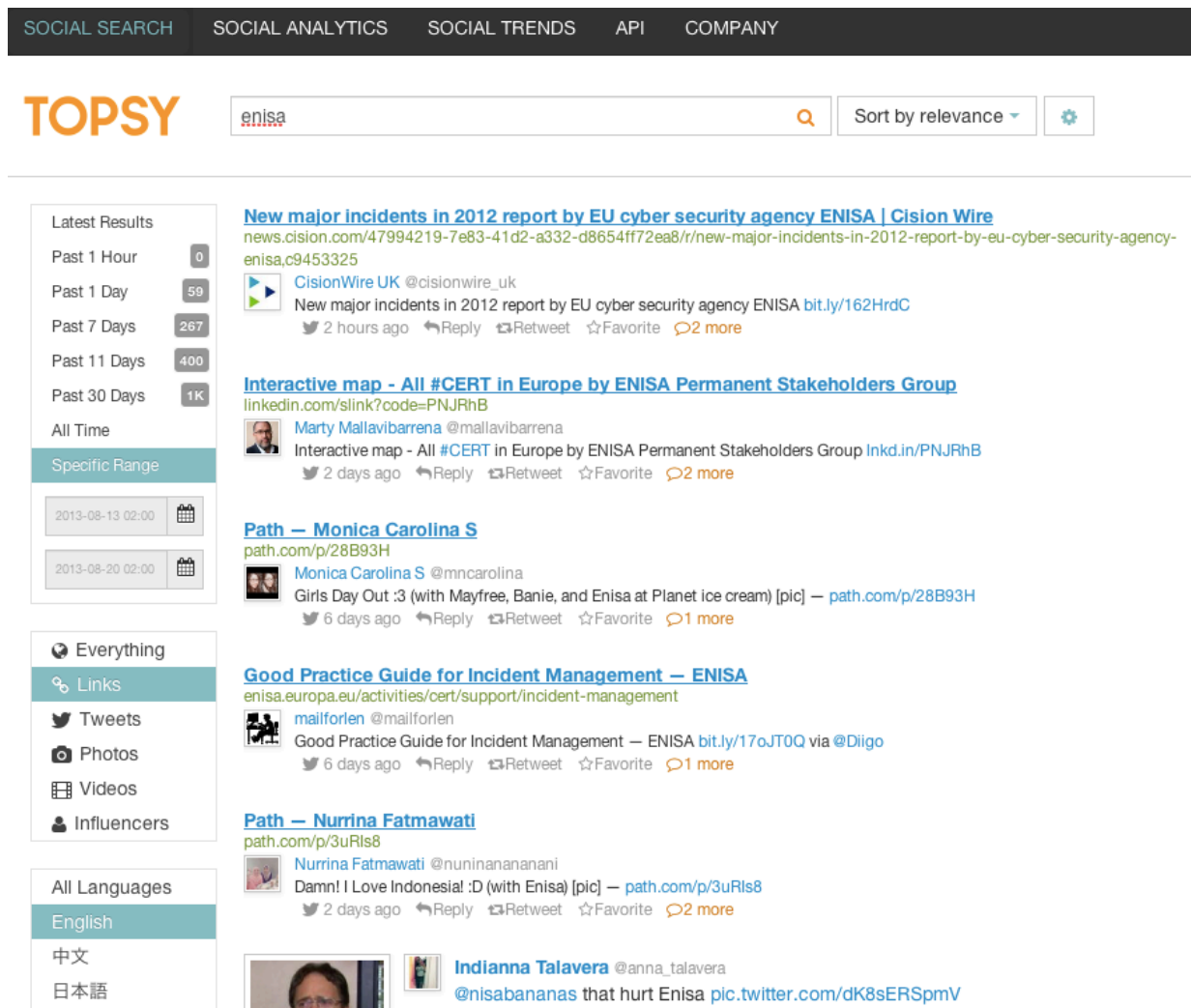
Category A – services which you use regularly in your CERT work;

Category B – services which you do not use regularly in your CERT work, but you 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.

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

For this task you can use the topsy.com service. You should put ‘ENISA’ keyword in the search window and choose a specific time range.



The screenshot shows the Topsy search interface. At the top, there are navigation tabs: SOCIAL SEARCH, SOCIAL ANALYTICS, SOCIAL TRENDS, API, and COMPANY. The search bar contains the keyword 'enisa'. The results are displayed in a list format with various filters on the left side.

**Filters:**

- Latest Results
- Past 1 Hour: 0
- Past 1 Day: 59
- Past 7 Days: 267
- Past 11 Days: 400
- Past 30 Days: 1K
- All Time
- Specific Range: 2013-08-13 02:00 to 2013-08-20 02:00
- Everything
- Links
- Tweets
- Photos
- Videos
- Influencers
- All Languages
- English
- 中文
- 日本語

**Search Results:**

- New major incidents in 2012 report by EU cyber security agency ENISA | Cision Wire**  
news.cision.com/47994219-7e83-41d2-a332-d8654ff72ea8/r/new-major-incidents-in-2012-report-by-eu-cyber-security-agency-enisa.c9453325  
CisionWire UK @cisionwire\_uk  
New major incidents in 2012 report by EU cyber security agency ENISA bit.ly/162HrdC  
2 hours ago
- Interactive map - All #CERT in Europe by ENISA Permanent Stakeholders Group**  
linkedin.com/slink?code=PNJRhB  
Marty Mallavibarrena @mallavibarrena  
Interactive map - All #CERT in Europe by ENISA Permanent Stakeholders Group Inkd.in/PNJRhB  
2 days ago
- Path – Monica Carolina S**  
path.com/p/28B93H  
Monica Carolina S @mncarolina  
Girls Day Out :3 (with Mayfree, Banie, and Enisa at Planet ice cream) [pic] – path.com/p/28B93H  
6 days ago
- Good Practice Guide for Incident Management – ENISA**  
enisa.europa.eu/activities/cert/support/incident-management  
mailforlen @mailforlen  
Good Practice Guide for Incident Management – ENISA bit.ly/17oJT0Q via @Diigo  
6 days ago
- Path – Nurrina Fatmawati**  
path.com/p/3uRls8  
Nurrina Fatmawati @nuninananani  
Damn! I Love Indonesia! :D (with Enisa) [pic] – path.com/p/3uRls8  
2 days ago
- Indianna Talavera @anna\_talavera**  
@nisabananas that hurt Enisa pic.twitter.com/dK8sERSpmV

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



As you will see such a general search can come up with a lot of an irrelevant information. Thus please tune the search by excluding false negative results as much as possible, e.g. these which include words like 'love', 'girls', 'Bukvic', 'Custovic'. Find the relevant search operator.

## 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.

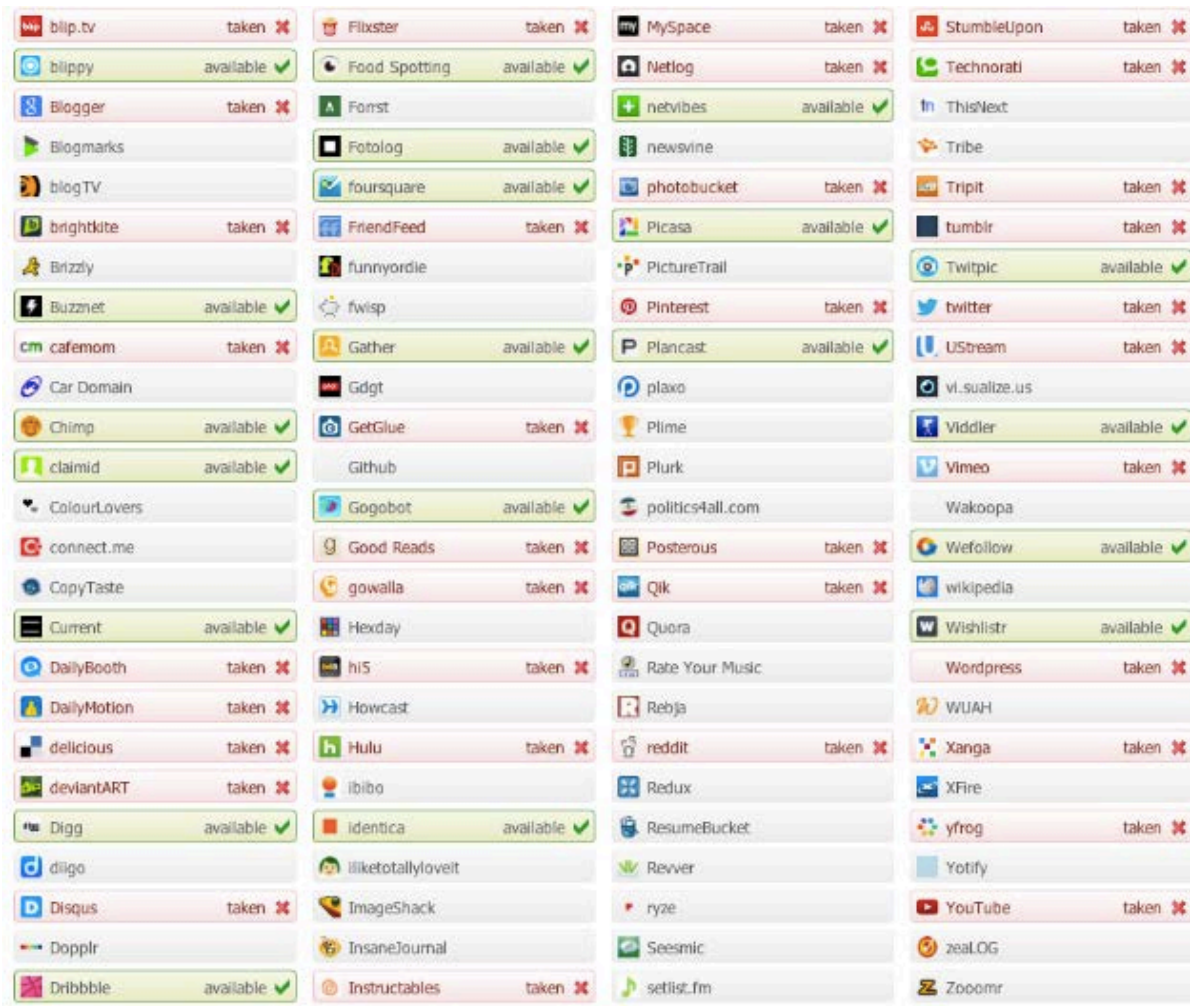


Figure 12: The result of the 'ENISA' search in the namechk.com service

Your additional task is to provide a list of such accounts which could be used for further monitoring.

## 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>.

After the installation the software interface shows the available functionality.

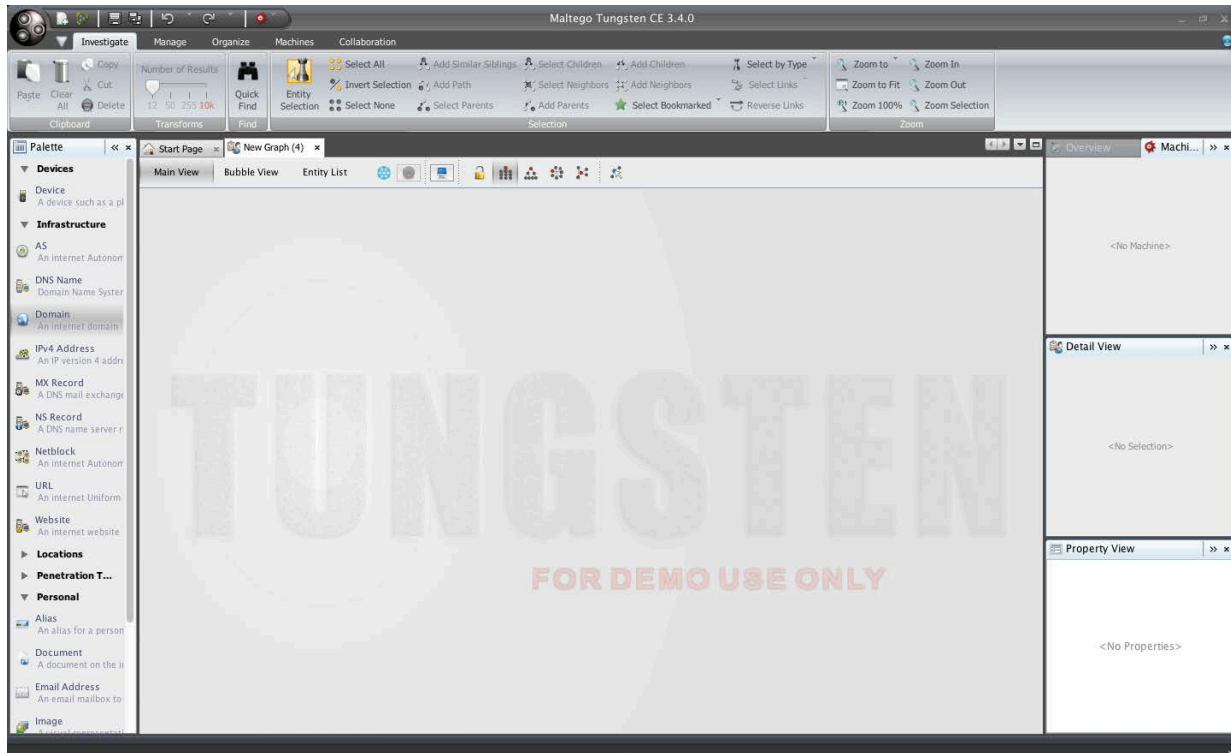


Figure 13: Screenshot of Maltego application

### 5.3.1 Finding PGP keys

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

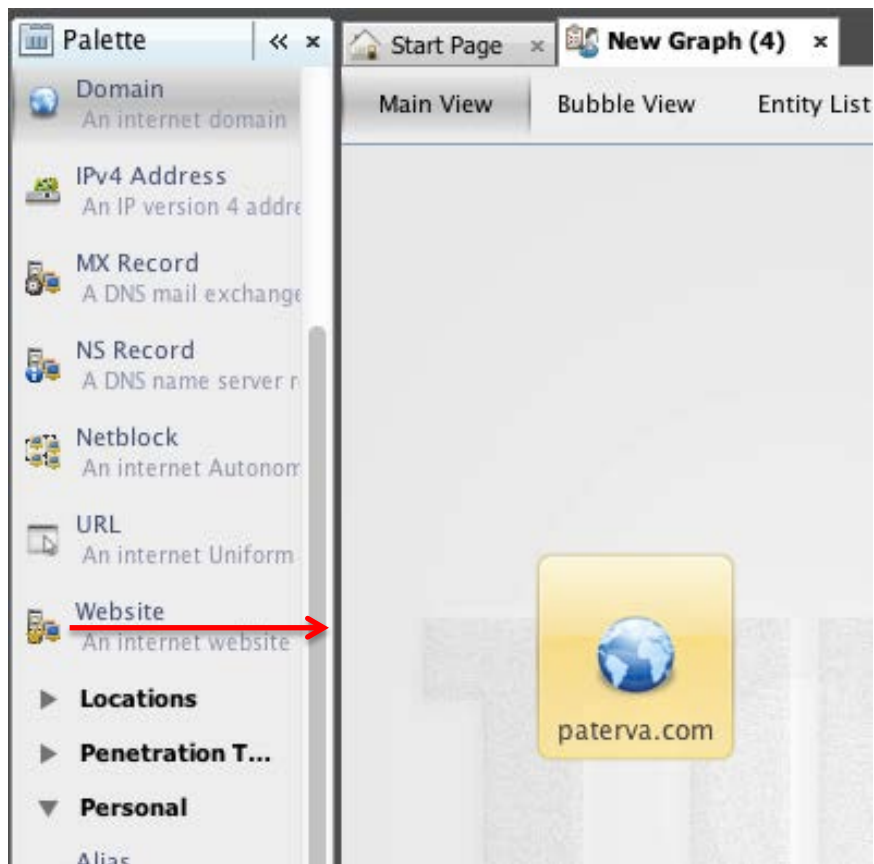


Figure 14: Choosing the organisation by website selection

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

Then your task is to receive information about particular PGP keys available in the organisation. Please find the appropriate function in the service to get it.

Your result should be the same or similar to that presented in the figure below.

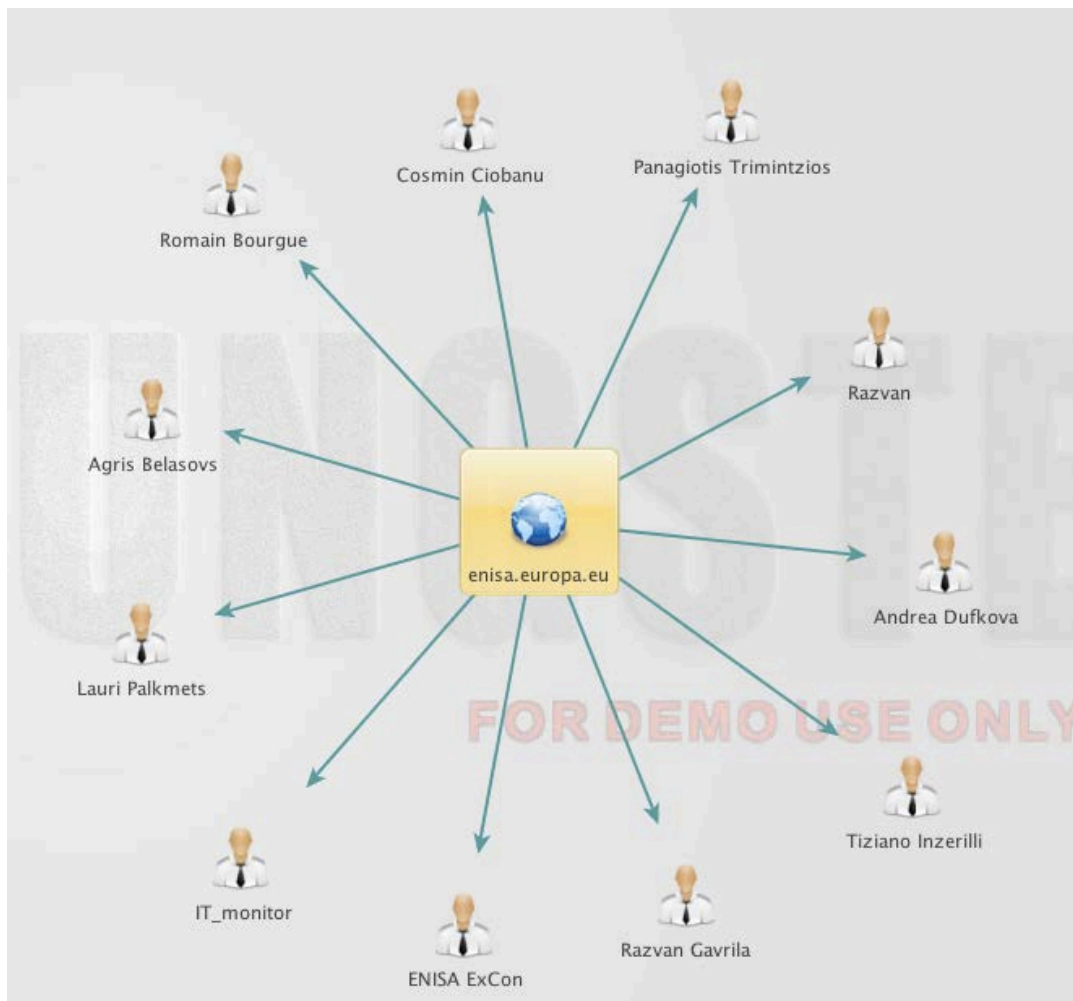


Figure 15: Screenshot 1 – Information about the PGP keys available in the organisation

### 5.3.2 Finding phone numbers

Your next task is similar, but now you should add information about the available phone. If you are successful, your result should be similar to that presented below.

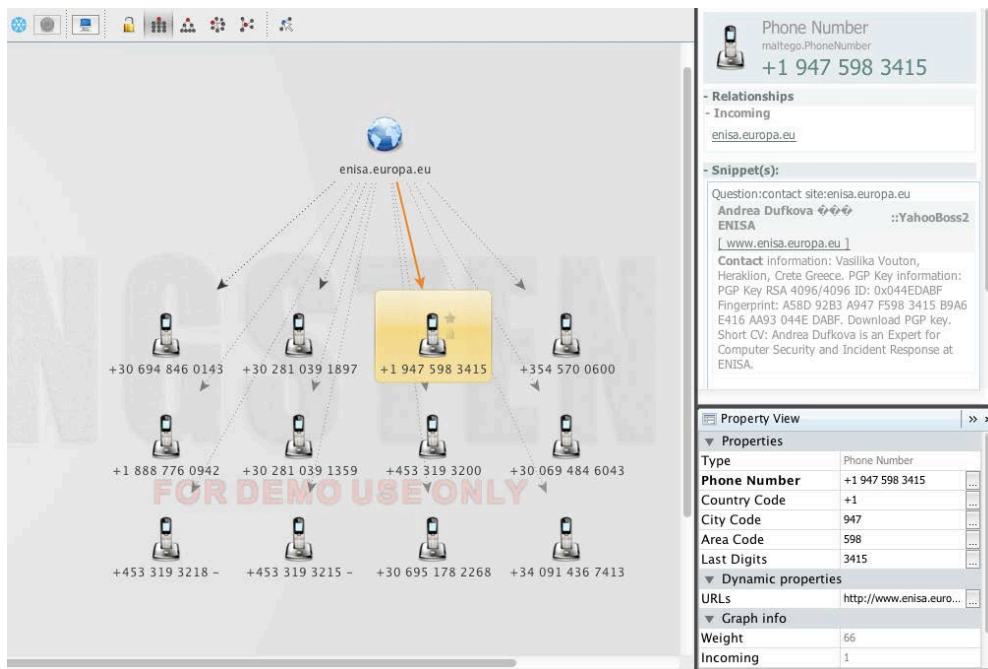


Figure 16: Screenshot 2 – Information about the organisation phone numbers with the property view of one of the numbers

The sub-tasks of Task 3 showed just a few examples of possible tracking activities. Maybe you know other known sources of valuable information? Always remember that you are permitted to use legal methods of information collection only.

The Maltego Tool it is a very powerful tool and only a small part of its capabilities was presented during the exercise. Now you can experiment with this tool and find other possible ways to use it. To learn about more its functionalities, watch the video tutorials prepared by Paterva. They are available at: <http://www.paterva.com/web6/documentation/index.php>.

## 6 Task 4 – Legal aspects of Internet monitoring services

In this part of the exercise your task is to develop a kind of internal legal guide for doing investigation work for your team. Consider potential risks in particular.

Think about:

- establishing relations with law enforcement agencies and:
- your legal power and real needs.
- the protection of collected information
- the protection of personal data if it will be part of your monitoring and could be made public as a result.

Now, analyse the laws of your country. The result of your analysis can be a matrix with name of your country (rows) and law rules (columns) – see example below.<sup>7</sup>

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 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			
[...]			

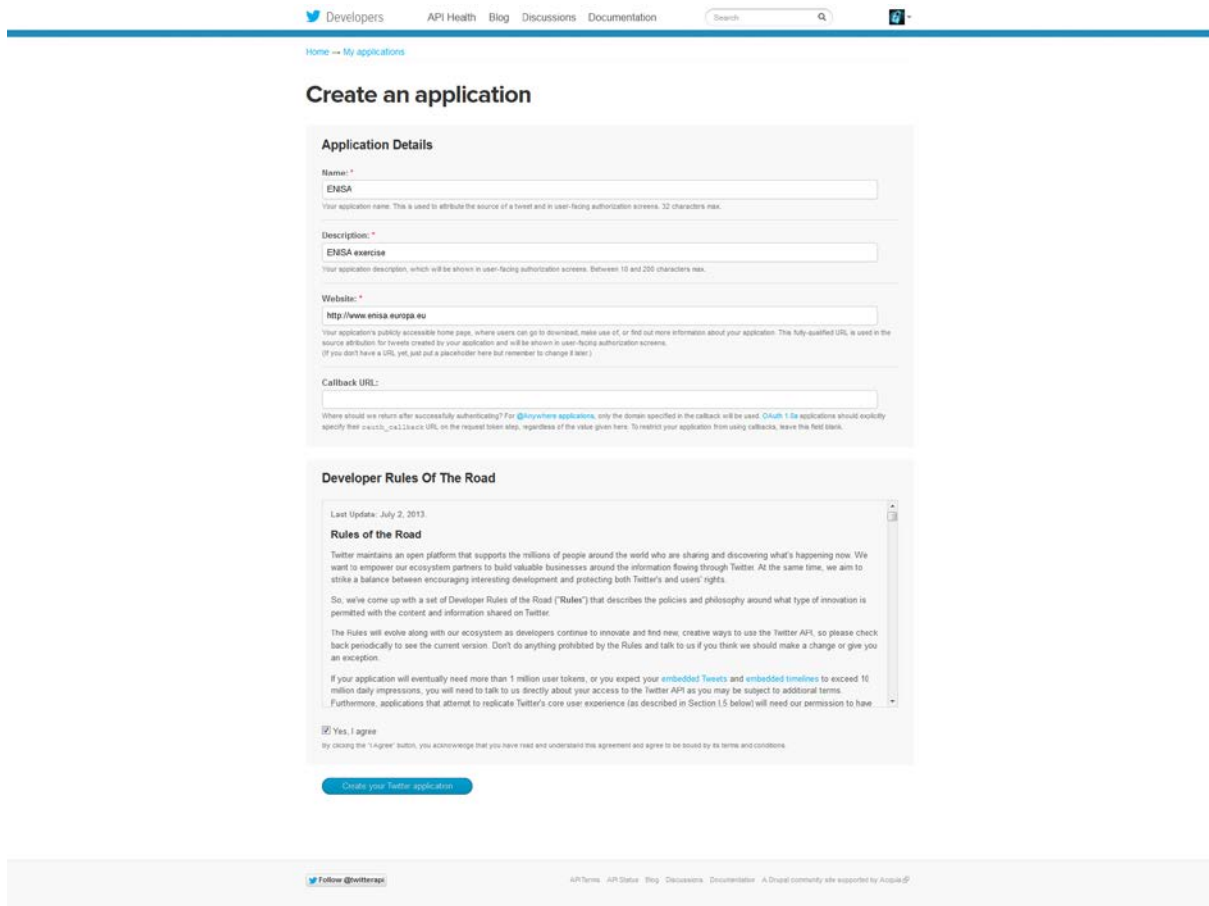
When you are ready present your findings and discuss with others. Pay special attention to the protection of personal data.

<sup>7</sup> The proposals in the table are not all examples from the law.

## 7 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.



The screenshot shows the Twitter developer website interface for creating a new application. At the top, there is a navigation bar with links for 'Developers', 'API Health', 'Blog', 'Discussions', and 'Documentation', along with a search bar and a Twitter logo. Below the navigation bar, the page title is 'Home → My applications' and the main heading is 'Create an application'.

The 'Application Details' section contains several input fields:

- Name:** A text input field containing 'ENISA'.
- Description:** A text input field containing 'ENISA exercise'.
- Website:** A text input field containing 'http://www.enisa.europa.eu'.
- Callback URL:** An empty text input field.

Below the input fields is the 'Developer Rules Of The Road' section, which includes a 'Last Update: July 2, 2013' and a 'Rules of the Road' heading. The text explains Twitter's commitment to an open platform and outlines the 'Developer Rules of the Road' that govern the use of the Twitter API. It mentions that developers must agree to these rules and that applications exceeding 1 million daily impressions or 10 million daily impressions may be subject to additional terms. At the bottom of this section, there is a checkbox labeled 'Yes, I agree' which is checked.

At the bottom of the form, there is a blue button labeled 'Create your Twitter application'.

The footer of the page includes a 'Follow @twitterapi' link, a list of links for 'API Terms', 'API Status', 'Blog', 'Discussions', and 'Documentation', and a note that the site is powered by Acquia.

Figure 17: Screenshot presenting the template for creating an application on the Twitter website





```
// number of last checked statuses
$nottweets = 10;
session_start();
// https://github.com/abraham/twitteroauth
require_once("twitteroauth/twitteroauth/twitteroauth.php");
// Twitter API
$consumerkey = "XXXXX";
$consumersecret = "XXXXX";
$access_token = "XXXXX";
$access_token_secret = "XXXXX";
function getConnectionWithAccessToken($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, $access_token,
$access_token_secret);
$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=" . $nottweets);
    // 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
```

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

```
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 availability of such API (<https://github.com/abraham/twitteroauth>). This helps to build a databases 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 . $turl . $separator . $tweettext . $separator;
            // add to alert array
            array_push($alertarray, $alert);
        }
    }
}
// check tweet
if (strpos($tweettext, $warningstring) !== false) {
```

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

## 8 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>
```

## 9 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 AnonSomewhere" 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 AnonSomewhere" 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 AnonSomewhere" 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 AnonSomewhere" 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 -!- L3JION [LEJION@1671671] has quit [Ping timeout: 121 seconds]  
--- Log closed Jun 29 01:22:25 2013





## 10 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](mailto:info@enisa.europa.eu)  
[www.enisa.europa.eu](http://www.enisa.europa.eu)