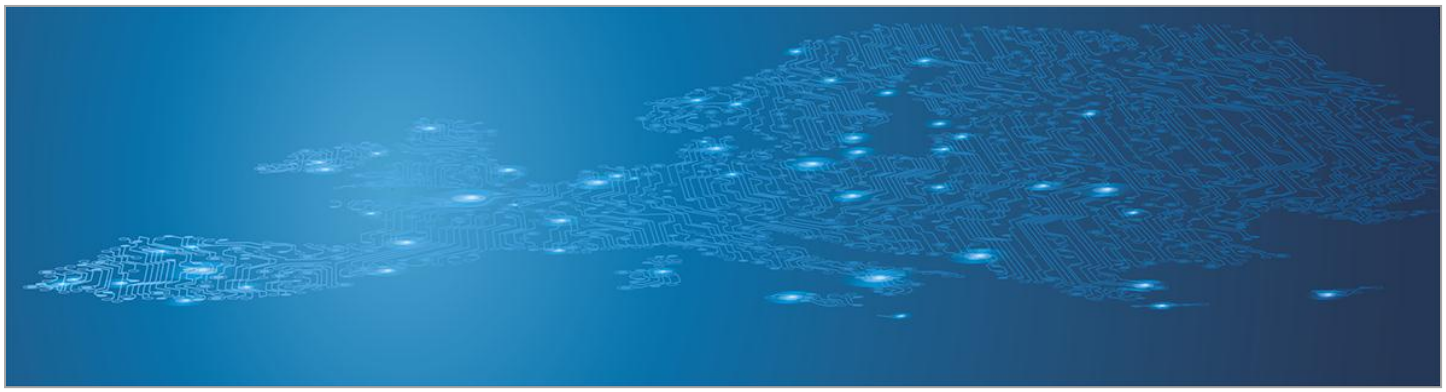

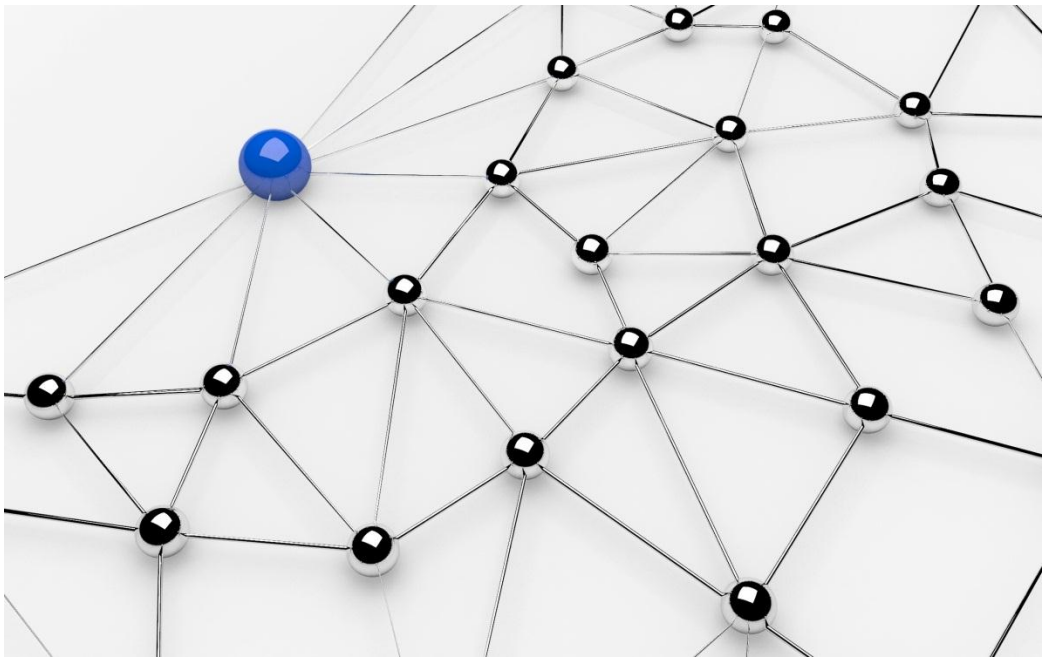

Presenting, correlating and filtering various feeds

Handbook, Document for teachers

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 and Alan Thomas Robinson from S-CURE, The Netherlands, Mirosław Maj, Tomasz Chlebowski, Paweł Weżgowiec 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:

1. Tomas Lima from CERT.PT

2. The countless people who reviewed this document.

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

Table of Contents

Main Objective	This exercise focuses on the technical aspects of using visualisation to present, correlate and filter various feeds. The scenario will also cover the organisational aspects. In this scenario the students will be part of the CERT for a fictitious organisation which is subject to cybercrime activities.	
Targeted Audience	CERT Technical specialists and staff who are responsible for presenting incident-related information. AbuseHelper exercise (14 Proactive Incident Detection[1]) is recommended as a prerequisite.	
Total Duration	~6.0 hours	
Time Schedule	Introduction to the exercise	1.0–2.0 hours
	Task 1: Identifying and configuring data feeds	2.0 hours
	Task 2: Using Kibana	0.5 hours
	Task 3: Tracing cybercrime activity	2.0 hours
	Summary of the exercise	0.5 hours
Frequency	Once per team	

[1] *ENISA Exercise Material* http://www.enisa.europa.eu/activities/cert/support/exercise

1 General Description

During this exercise the trainer will introduce the students to visualisation with Kibana,[2] used as a primary solution (the underlying AbuseHelper[3] that provides input data feeds, should be familiar to students from the prerequisite exercise 14 Proactive Incident Detection).

The students will be organised in groups of three (the technical equipment must reflect this requirement) and will take on typical IRT roles (Analyst, Manager, Technician[4]). Each team will form the CERT for a fictitious organisation subject to business process related cybercrime activities.

At the beginning the trainer will introduce the data/security visualisation, while presenting the considerations on why and how the system will be set up and what information should and would be presented. The trainer will introduce several different solutions offering students a chance to interact with each other.

During the first part of the exercise the teams will search for data feeds (from a given list) relevant to the organisation and its specific threat model. These data feeds have to be configured in AbuseHelper to deliver information to be visualised in Kibana.

In the second part, the students should analyse the architecture of the visualisation tools (Logstash,[5] Elasticsearch[6] and Kibana) and get familiar with the capabilities and limitations of each component.

In the third part the trainer will provide the teams with a set of cybercrime scenarios which should be traced and identified using Kibana.

Each team should write a report about the findings and the process of analysing the data with Kibana and its advantages and disadvantages. These reports will be presented and discussed during the summary.

[2] *Kibana http://kibana.org/*
[3] *AbuseHelper http://www.abusehelper.be/*
[4] *See ENISA exercise No.3 Recruitment of CERT staff*
[5] *Logstash http://www.logstash.net/*
[6] *Elasticsearch http://www.elasticsearch.org/*

2 Exercise Course

'Information visualisation is the study of (interactive) visual representations of abstract data to reinforce human cognition. The abstract data include both numerical and non-numerical data, such as text and geographic information.'[7]

Information visualisation provides the user with the capability to 'see' and analyse large amounts of data using an intuitive approach instead of trying to read single lines of text. This text is parsed, matched to categories and represented in a sensible way. The user learns how to identify a baseline of usual behaviour and detect anomalies by seeing the patterns in the visualised data.

Different ways to present visualised data are described later in this exercise.

2.1 AbuseHelper

The trainer starts the introduction with a summary of exercise No. 14 (Proactive Incident Detection, PID):

a) Definition of proactive incident detection

Proactive detection of incidents is the process of discovery of malicious activity in a CERT's constituency through internal monitoring tools or external services that publish information about detected incidents, before the affected constituents become aware of the problem. It can be viewed as a form of early warning service from the constituents' perspective. Effective proactive detection of network security incidents is one of the cornerstones of an efficient CERT service portfolio capability. It can greatly enhance a CERT's operations, improve its situational awareness and enable it to handle incidents more efficiently, thus strengthening the CERT's incident handling capability, which is one of the core services of national/governmental CERTs.

b) Introduction and presentation of information feeds

Most external services offer incident data in the form of IP addresses, URLs, domains or malware associated with a particular malicious activity, such as a bot, C&C server, malicious URL or scanning. Sometimes more sensitive data are offered, such as stolen user credentials or credit card data. Some services may also offer alerts in more abstract forms depending on detection models used internally in the service.

c) Explanation of technology used in AbuseHelper (XMPP)

The software was developed to provide a framework on which incident response teams can build their own handling systems and processes. It is developed under the terms of the MIT licence to let teams take part in the progress of the system.

AbuseHelper is written in Python and developed relying on XMPP protocol (not mandatory) and agents. The base principle is to control an agent via a central chat room where all bots are listening. Agents are exchanging information in subrooms. AbuseHelper is then scalable and each agent follows a KISS (Keep it Simple, Stupid*) approach. Each user is able to produce the perfect workflow for his business. The user just needs to take the agents he needs and connect them together.*

In the references section and in the folder on the Virtual Image (references /usr/share/trainer/14_PID), additional material is provided, which will help you in addressing this part of the exercise.

[7] *Information visualisation* http://www.cs.ubc.ca/labs/imager/tr/2008/pitfalls/pitfalls.pdf

3 Tools and data overview

The focus in the introduction (and in the complete exercise) will be on the first and the last step in the data flow.

The following graph shows the components taking part in the exercise, their function and the way data is processed. Use this figure to make yourself familiar with the data flow and as a reminder of the role each the applications has.

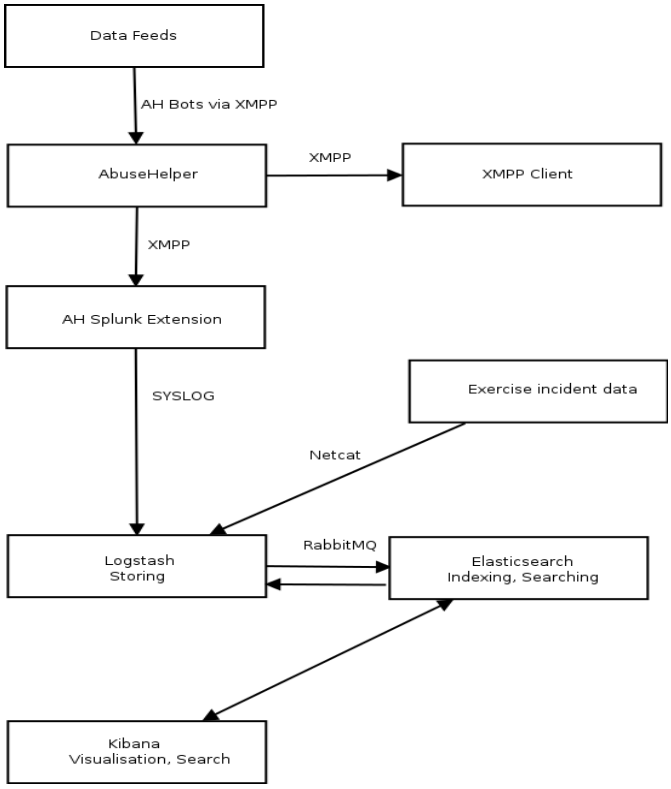

Figure 1: Data flow and components

3.1 AH LogCollector[8]

Tomas Lima from CERT.PT has provided an extension to send AbuseHelper data to a syslog compatible listening service.

The extension connects as a XMPP client to the abusehelper.sources room, reads all incoming data and sends it to a configured TCP port using the SYSLOG-NG protocol.

It is implemented in Python, has no special licence and is easy to set up. It writes a log file of its actions in the file abushelperextension.log located in the directory where it is run (see below under 'Visualisation tools').

[8] https://bitbucket.org/certpt/logcollector-abusehelper-extension/overview

3.2 AH Genericevent[9]

Tomas Lima also has provided this bot to AbuseHelper. It helps in writing sanitiser scripts to normalise the data collected by AbuseHelper bots before delivering it to external analysis tools (like Splunk, Logstash). Installation commands can be found in section 'Starting the environment'.

It is implemented in Python and has no special licence. The code ships with example for sanitiser scripts which can be used to build custom scripts for your own needs.

3.3 Logstash

Logstash[10] is a tool for managing events and logs. It is used to collect logs in various ways. In this scenario it is configured to listen for incoming data, parse it to create categories which are used by Kibana and hand it over to Elasticsearch for indexing and analysis.

Logstash is a Java application under the Apache 2.0 licence.

3.4 Elasticsearch

Elasticsearch[11] processes and analyses the data. It provides the backend functionality used by Kibana for data handling (like searching).

Elasticsearch implements the Apache Lucene[12] query syntax in Java and is licensed with the Apache 2.0 licence.

3.5 Kibana[13]

Kibana is user interface presenting data processing functions.

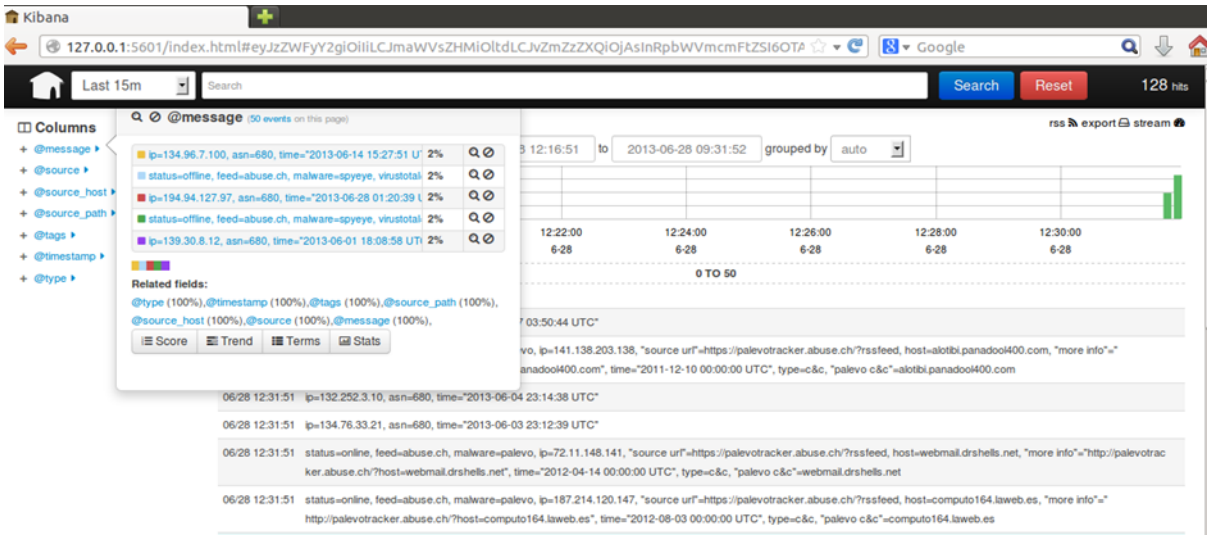

Figure 2: Kibana UI screenshot

The configuration mainly consists of including the IP and Port of your Elasticsearch deployment. Afterwards it can be started with the command 'ruby kibana.rb'. Kibana is developed in Ruby and is

[9] https://bitbucket.org/certpt/genericevent-abusehelper
[10] http://logstash.net/
[11] http://www.elasticsearch.org/
[12] https://lucene.apache.org/core/3_5_0/queryparsersyntax.html
[13] http://kibana.org/

licensed 'without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.'[14]

[14] *Taken from the LICENSE.md file of the installation source*

4 Alternative solutions

There are a multitude of solutions and implementations regarding the visualisation of large data amounts. Approaches include Graphs,[15] Parallel Coordinates and Heatmaps.[16]

These methods will be described below and used in the exercise introduction to impart a broader perspective on visualisation to the students. For each visualisation approach one implementation will be presented to give some hints how to use it.

4.1 Parallel Coordinates

'**Parallel coordinates** is a common way of visualizing high-dimensional geometry and analysing multivariate data.

To show a set of points in an n-dimensional space, a backdrop is drawn consisting of *n* parallel lines, typically vertical and equally spaced.[17]

4.1.1 Picviz

'Picviz is a parallel coordinate's plotter which enables easy scripting from various inputs (tcpdump, syslog, iptables logs, apache logs, etc.) to visualise your data and discover interesting results quickly.'

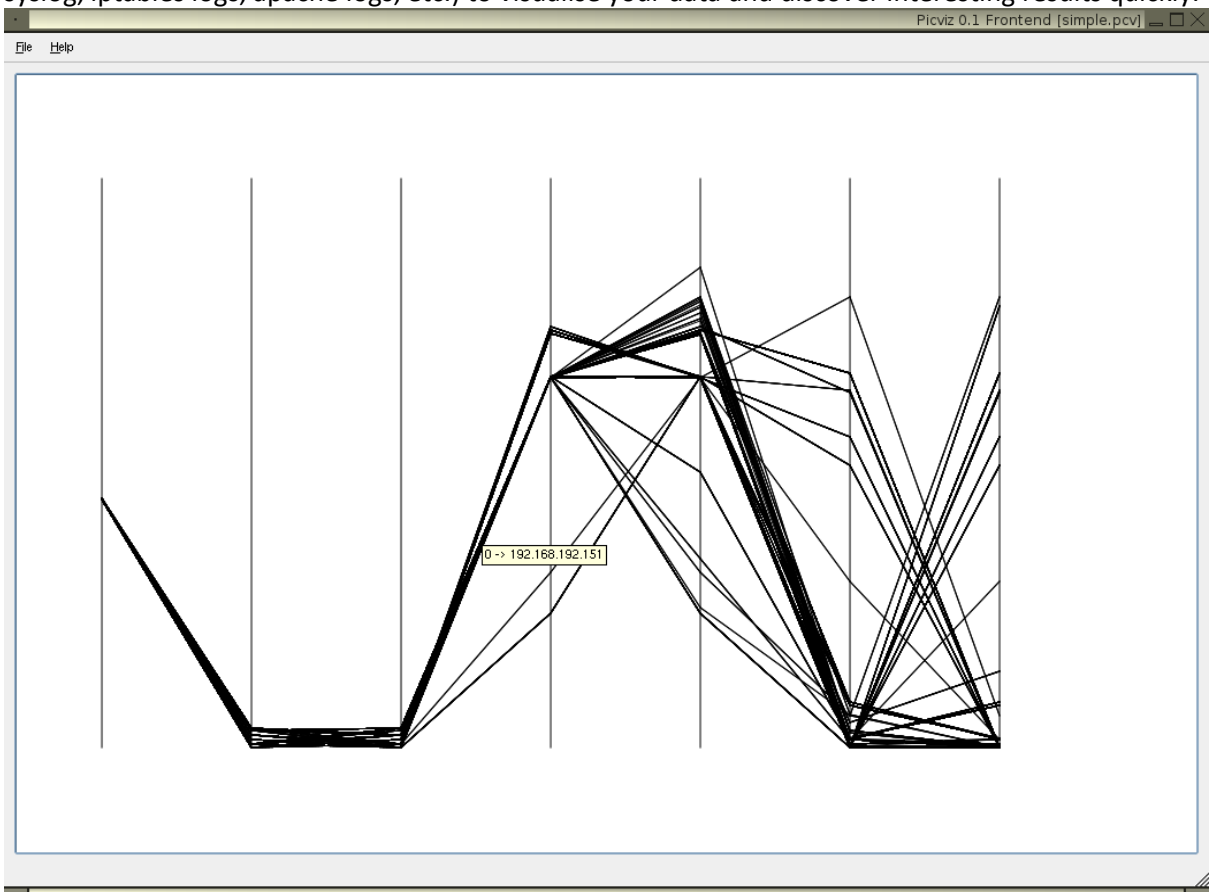

Figure 3: Picviz[18]

[15] *Graphs* https://en.wikipedia.org/wiki/Graph_%28data_structure%29
[16] https://en.wikipedia.org/wiki/Heatmap
[17] https://en.wikipedia.org/wiki/Parallel_coordinates
[18] *Picviz* http://commons.wikimedia.org/wiki/File:Picviz-0.1.png

4.2 Heatmaps

'A heat map is a graphical representation of data where the individual values contained in a matrix are represented as colours.'[19]

4.2.1 Team Cymru Heatmap[20]

The map represents individual class A networks with the blue dots indicating malicious activity originating from the network.

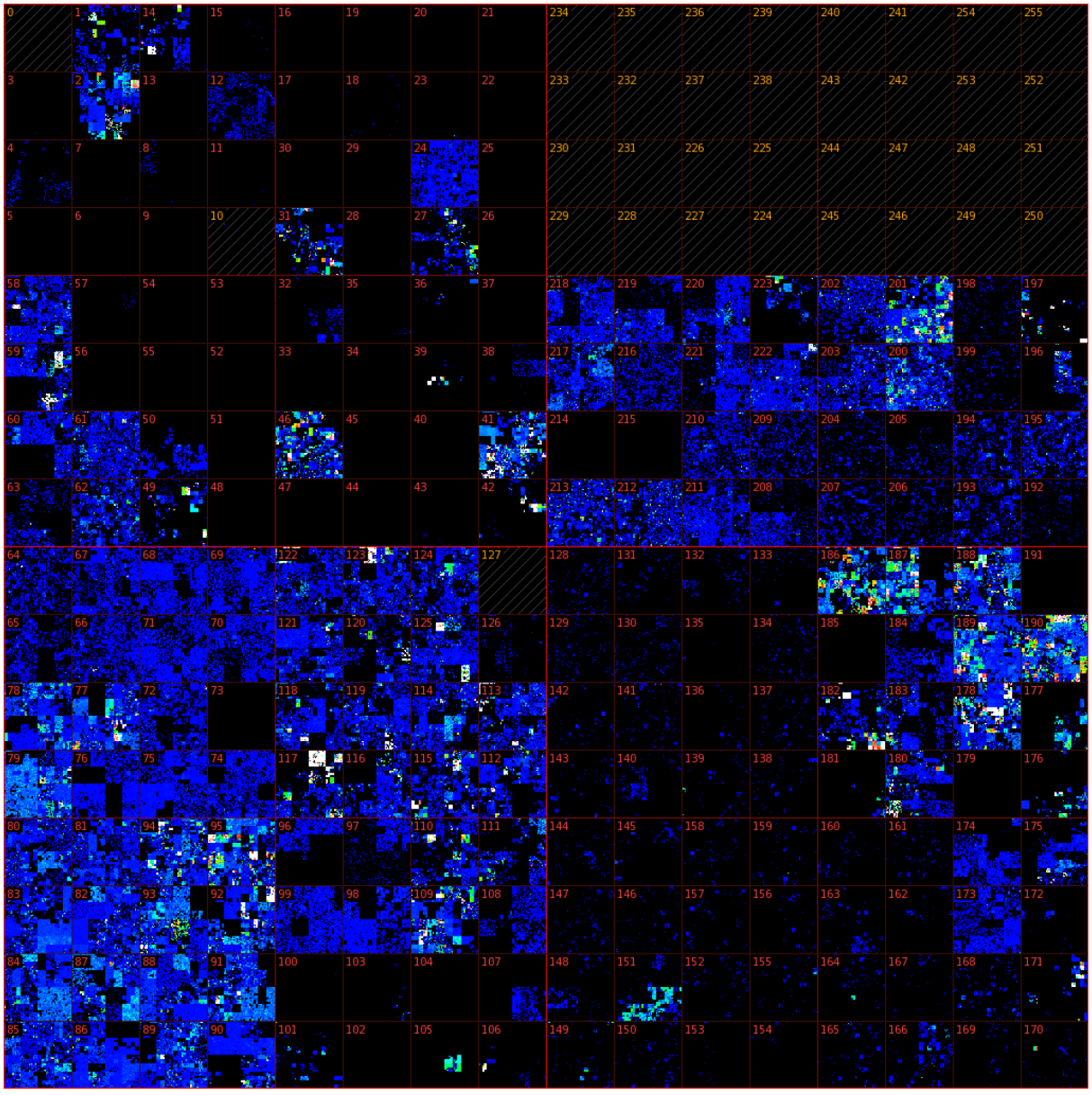

Figure 4: Team Cymru Heatmap[21]

[19] https://en.wikipedia.org/wiki/Heat_map
[20] https://team-cymru.org/Monitoring/Malevolence/maps.html
[21] https://team-cymru.org/stats/hilbert/full.png

4.3 Graph visualisation[22]

'Graph drawing is an area of mathematics and computer science combining methods from geometric graph theory and information visualisation to derive two-dimensional depictions of graphs arising from applications such as social network analysis, cartography, and bioinformatics.'[23]

4.3.1 ProcDOT[24]

'This tool processes Sysinternals Process Monitor (Procmon) logfiles and PCAP-logs (Windump, Tcpdump) to generate a graph via the GraphViz suite. This graph visualises any relevant activities (customisable) and can be interactively analysed.'[25]

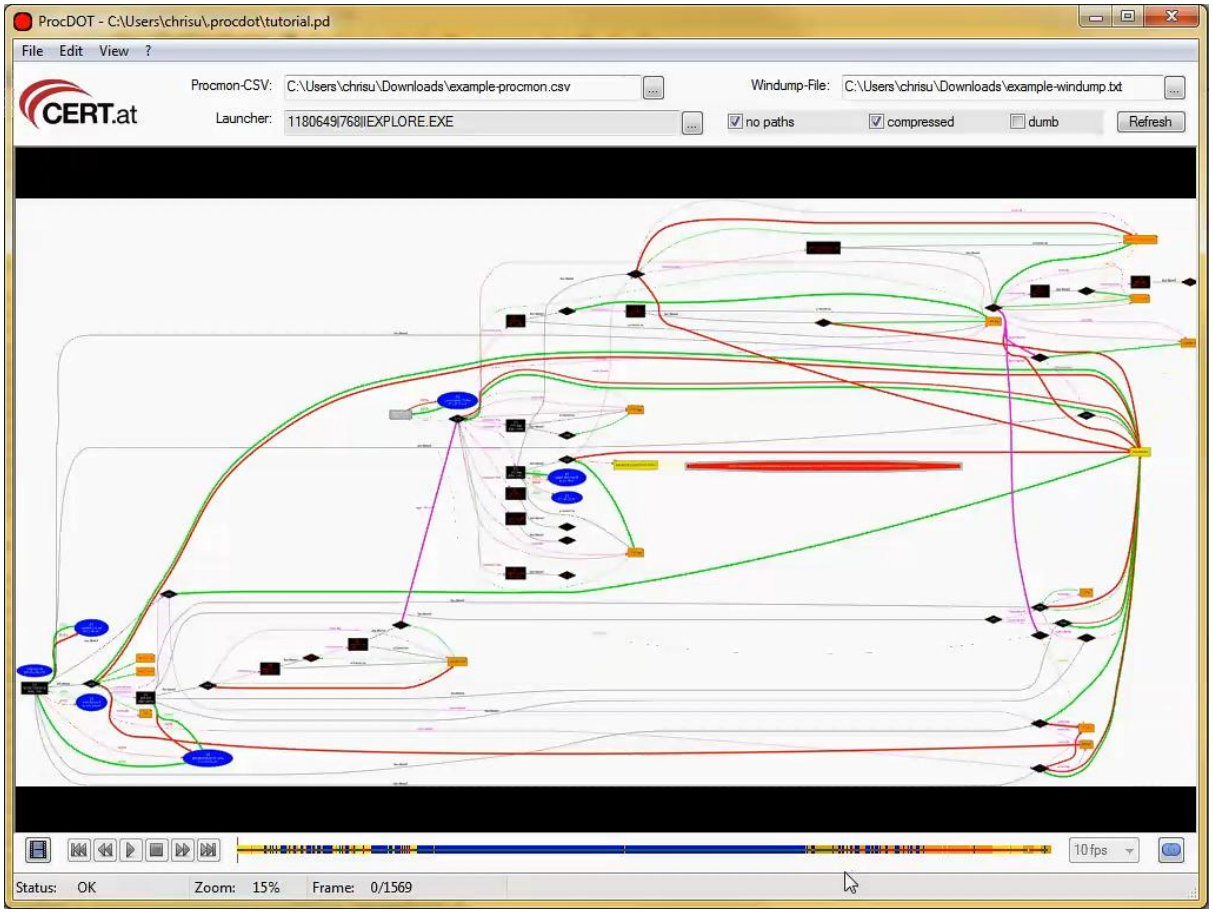

Figure 5: Procdot[26]

[22] https://en.wikipedia.org/wiki/Graph_visualization
[23] https://en.wikipedia.org/wiki/Graph_visualization
[24] http://cert.at/downloads/software/procdot_en.html
[25] http://cert.at/downloads/software/procdot_en.html
[26] http://cert.at/downloads/software/procdot_en.html

5 Introduction to the scenarios

5.1 Organisation

Although this is not meant to be a full-scale role-play exercise there is some background information which should help the students to understand the data to be analysed and the context of the incidents.

The students will form the CERT of a targeted organisation. In this part the trainer will describe the organisation, its business processes and the profile of possible (financially motivated) attackers.

Background information regarding the context of the data:

The company in question is a provider for e-commerce solutions. It hosts the business infrastructure (site, shop solution, payment) for its customers in a software as a service (SaaS)[27] model. The sites it hosts generate daily revenue of approximately €10 million.

Technical details of the technical infrastructure of the company:

- Domain: example.com
- Network: 192.0.32.0/20
- Autonomous System Number (ASN)[28]: AS26711
- Apache Webserver for administrative login
- ModSecurity intrusion detection system (IDS)

This information is essential to the students to filter the incoming data in Kibana and identify the incidents related to the company.

5.2 Starting the environment

Before the practical part of the exercise can be started, the environment has to be prepared.

5.2.1 AbuseHelper

There are three ways to approach the exercise:

1. Exercise 14 Proactive Incident Detection has been run beforehand and the AbuseHelper installation is still available. In this case you can go forward to the 'AbuseHelper Genericevent extension' section below.

2. You can also run the exercise without an AbuseHelper installation and use the static data (located in the event_$X folders in the exercise file system) files for all parts. If you choose this approach, go forward to the 'Visualisation tools' section below.

3. AbuseHelper will be installed and configured during the class

In this case the following installation and configuration steps are necessary (more information can be found in the handbook of exercise 14 Proactive Incident Detection):

The Virtual Image has been prepared with all necessary tools, libraries and sources to install and use AbuseHelper. Please be aware that an Internet connection is mandatory to make use of the live feeds.

[27] *Software as a Service (SaaS)* http://cloudtaxonomy.opencrowd.com/taxonomy/software-as-a-service/
[28] *Autonomous System (AS) Numbers* http://www.iana.org/assignments/as-numbers/as-numbers.xhtml

1. Ejabber Daemon

sudo /etc/init.d/ejabberd start # Start the Jabber service
sudo ejabberdctl register abusehel localhost exercise # register a user for the bots (username host password)
sudo ejabberdctl register trainer localhost exercise # register a user for the trainer (username host password)
sudo ejabberdctl register trainee localhost exercise # register a user for the students (username host password)
sudo vi /etc/ejabberd/ejabberd.cfg # open the ejabberd configuration file and edit the following lines ***max_user_sessions 100*** # maximum sessions for a single user
s2s_default_policy deny # deny server to server communication
%%
{shaper, c2s_shaper}, # search for and comment out the default shaper configuration
{mod_muc,
"conference.@HOST@"},
[
%%{host,
{access, muc},
{access_create, muc},
{access_persistent, muc},
{access_admin, muc_admin},
{max_users_admin_threshold, 20}, # add this entry
{max_user_conferences, 1000}, # add this entry
{max_users, 500}
]},
sudo /etc/init.d/ejabberd restart # restart ejabberd server

2. AbuseHelper

sudo useradd -m abusehel # add a system user for AbuseHelper
sudo mkdir -p /var/lib/ah2 # create the working directory
sudo chown root:abusehel /var/lib/ah2 #ownership of the working directory
sudo chmod 0750 /var/lib/ah2 # directory access rights set to read, write
cd /usr/share/trainer/14_PID/adds/abusehelper/ # change your current directory (trainee for the students)
sudo python setup.py install # run the AbuseHelper setup script

cd /usr/local/lib/python2.7/dist-packages/abusehelper # change directory
sudo python contrib/confgen/confgen.py /var/lib/ah2/production # start the configuration script

Enter the following information:

XMPP username: abusehel@localhost # as defined during user registration
XMPP password: exercise # you will be asked to enter this twice
XMPP lobby channel: abusehelper # this is the initial channel to connect to when starting the Jabber client
Configure mailer? No # do not let AbuseHelper send alert mails
sudo chown -R root:abusehel /var/lib/ah2/production # access rights have to be corrected after the configuration script
sudo chmod 0750 /var/lib/ah2/production # see above
sudo chmod g+w /var/lib/ah2/production/archive # see above
sudo chown abusehel /var/lib/ah2/production/log # this directory has been added and must be owned by the abusehel system user for logging
sudo chown abusehel /var/lib/ah2/production/state # see above
sudo vi /var/lib/ah2/production/startup.py # open the startup script and check the entries made by means of the confgen script
Insert this line after 'service_room=service_room,' in the 'def basic' section:
xmpp_ignore_cert = True, # this deactivates checking ssl certificates
Comment out the following line in the 'def configs' section:
yield basic("roomgraph")

3. Start AbuseHelper

sudo su - abusehel -s /bin/bash # change to the abusehel system user
botnet start /var/lib/ah2/production # start the bots defined in the startup.py script
botnet status /var/lib/ah2/production # ask for the status, at least one instance should be running
botnet stop /var/lib/ah2/production # stop the abusehelper bots

Logs can be found in these directories:

/var/lib/ah2/production/log/
/var/log/ejabberd/

To enable logging functionality for every bot (logs can be found from **/var/lib/ah2/production/log**) uncomment the corresponding lines.

5.2.2 Abusehelper Genericevent Extension

The generic event extension provides sanitiser scripts and functionality. It has to be installed and configured after AbuseHelper is running:

cd /usr/share/trainee/28_VCT/add/certpt-genericevent-abusehelper/

```
sudo cp -a contrib/enrichment /usr/local/lib/python2.7/dist-packages/abusehelper/contrib/
sudo chown root:staff -R /usr/local/lib/python2.7/dist-packages/abusehelper/contrib/enrichment
sudo chmod 2755 -R /usr/local/lib/python2.7/dist-packages/abusehelper/contrib/enrichment
sudo cp -a custom/* /var/lib/ah2/production/custom
sudo chown root:abusehel -R /var/lib/ah2/production/custom/
sudo chmod 0774 -R /var/lib/ah2/production/custom/
```

5.2.3 Visualisation tools

After AbuseHelper is running, the services dedicated to this exercise have to be started. All of them are preconfigured on the virtual image, so simple commands are sufficient:

1. ElasticSearch

```
sudo /usr/share/elasticsearch/bin/elasticsearch -p /var/run/elasticsearch.pid \ -Des.default.config=/etc/elasticsearch/elasticsearch.yml \ -Des.default.path.home=/usr/share/elasticsearch \ -Des.default.path.logs=/var/log/elasticsearch \ -Des.default.path.data=/var/lib/elasticsearch \ -Des.default.path.work=/tmp/elasticsearch \ -Des.default.path.conf=/etc/elasticsearch
```

2. Logstash:

```
sudo java -jar /usr/share/logstash/logstash.jar agent -f /etc/logstash/syslog.conf &
```

3. AH Logcollector (wait for one minute after starting logstash and check with '**netstat -ant|grep 5544**' if logstash is active, only if you installed and configured AbuseHelper):

```
cd /usr/share/trainer/28_VCT/add/certpt-logcollector-abusehelper-extension && python runner.py &
```

4. Kibana:

```
cd /usr/share/trainee/28_VCT/add/Kibana/ && ruby kibana.rb &
```

5. Restart the botnet (only if you installed and configured AbuseHelper):

```
sudo su – abusehel -s /bin/bash -c 'botnet restart /var/lib/ah2/production'
```

6. Start a browser (e.g. Firefox) and connect to http://localhost:5601/ you should see the following screen:

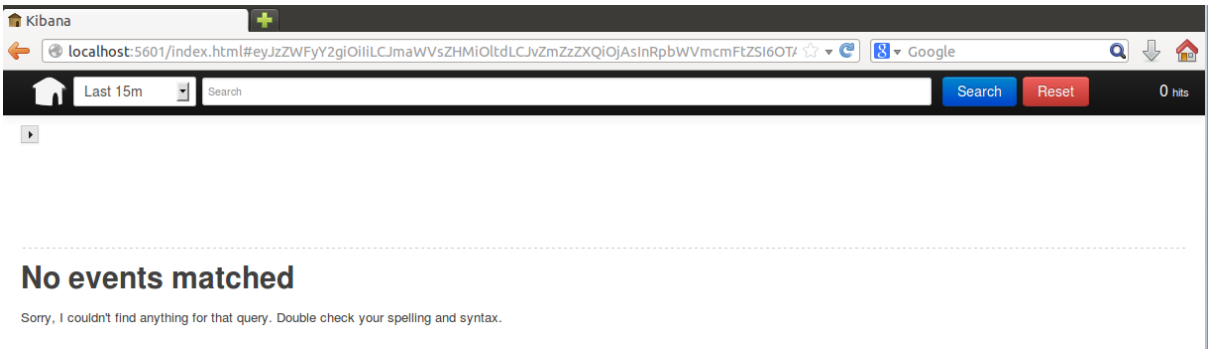

Figure 6: Kibana starting screen

In the top bar of the site you will see (from left to right):

- Home button;
- Pull-down menu to choose the timeframe (predefined and custom);
- Text field for search terms (executing it will filter the data), buttons to execute the search (filter) and reset the filter;
- Number of the lines of data matching the filter (50 lines will be shown in the main section of the page).

Figure 7: Kibana start screen with data

If data has been processed you will see an additional navigation menu on the left side.

The items in this list will change according to the displayed/filtered data. You can use each field to apply further actions (trending, filtering, and statistics) on the data in the main section.

In the main part (top right) are three links:

- rss: Use this one to import the displayed data into a newsreader of your choice, including using a Livefeed functionality;
- export: Export and save the data in CSV format;
- stream: Autoupdate the display of incoming data matching the search conditions.
- Beneath these is a timeline with options to select the frame and grouping (by time).
- At the bottom the data is shown (50 lines). If you select an event by clicking on it, the data will be shown separated into the fields:

Field	Action	Value
@fields.asn_code		N/A
@fields.asn_name		N/A
@fields.bgp		N/A
@fields.country_code		N/A
@fields.event_descr		N/A
@fields.event_hash		f0d87c0e9dcf094a3169c7f31055a1aea0e6d0f02e206588d03f90aa07707cf3
@fields.event_notes		N/A
@fields.event_type		N/A
@fields.host		N/A
@fields.incident_class		Phishing
@fields.incident_type		N/A
@fields.ip		202.74.46.90
@fields.netblock		N/A
@fields.register_time		2013-08-06
@fields.source_desc		N/A
@fields.source_name		CleanMX
@fields.status		N/A
@fields.trigger_time		2013-08-06
@fields.url		http://highlandsurplus.com/toolshack/images/.../scripts/example/remaxca/index.htm
@message		status=N/A, event_hash=f0d87c0e9dcf094a3169c7f31055a1aea0e6d0f02e206588d03f90aa07707cf3, asn_code=N/A, register_time=2013-08-06 15:21:29 UTC, source_name=CleanMX, event_type=N/A, url=http://highlandsurplus.com/toolshack/images/.../scripts/example/remaxca/index.htm, ip=202.74.46.90, trigger_time=2013-08-06 09:42:05 UTC2013-08-06 09:42:05, source_desc=N/A, country_code=N/A, bgp=N/A, host=N/A, incident_type=N/A, netblock=N/A, event_descr=N/A, incident_class=Phishing, asn_name=N/A, event_notes=N/A
@source		tcp://127.0.0.1:39171/
@source_host		127.0.0.1
@source_path		/
@tags		

Figure 8: Kibana single event details

5.2.4 Data to be analysed

The data will be injected into the current stream using netcat[29] and prepared logfiles. There are three separate files (one for each scenario). Injecting them can be done remotely (if the study environment permits) or locally via a simple shell command, eg: ***'netcat localhost 5544 < event_logs.txt'***.

The task related data files contain crafted information claiming to be from publicly available data feeds like Phishtank, Malware Domain List, Shadowserver, RSS feeds etc. and private information sources (Webserver Logs, IDS alerts).

5.3 Task 1 Evaluating data sources

For this task each student group needs Internet access to do the research.

The trainer hands out a form with available data feeds (see template AH-data-feed-eval.xls in the main folder). Each student group separately has to research the information to fill in the sheet and decide on which feeds would be useful for cybercrime tracing and explain the decision. These will be discussed at the end of Task 1.

[29] *Netcat* http://en.wikipedia.org/wiki/Netcat

Nr.	Feed Name	Short description	URL	License	Register	Cybercrime	Explanation
1	Dshield						
2	Shadowserver						
3	AbuseCH						
4	ARF						
5	Tailbot						
6	Arbor						
7	CleanMX						
8	Dragon Research						
9	IP Lists						
10	Malc0de						

Figure 9: AH evaluation spreadsheet

5.4 Task 2 Using Kibana

For this task Internet access is useful to be able to use live data for the analysis. If this is not possible the data files prepared for Task 3 can be used instead.

The students start the technical infrastructure with a fixed set of data feeds and get familiar with the UI of Kibana. This includes filtering, correlating, using the stream frontend and searching for a combination of terms. They also have to check that they see all relevant data.

The following feeds have been configured:

- AbuseCH;
- CleanMX;
- Malware Domainlist.

The students should answer the following questions and describe the approach they used to identify the related information:

- Which feeds are configured?
- Which ASN's are in the incoming data?
- Which ASN's are distributing malware?
- Which three ASN's have the most phishing incidents?

5.4.1 Filtering with Kibana

There are primarily two ways to filter data in Kibana. First through use of the auto generated (from the detected fields in the data) list at the left, and secondly through entries of search terms in the text field at the top.

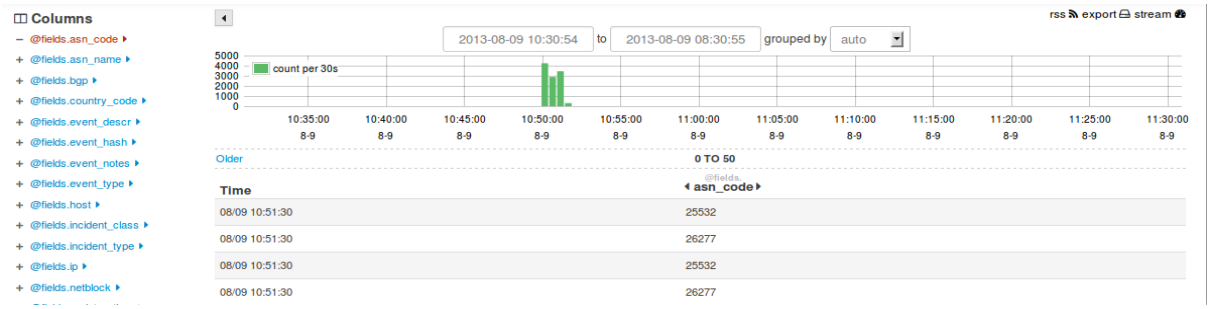

Figure 10: Kibana filter screenshot 1

You can interact with the list on the left in several ways. Clicking on the symbol (+/-) in front of the

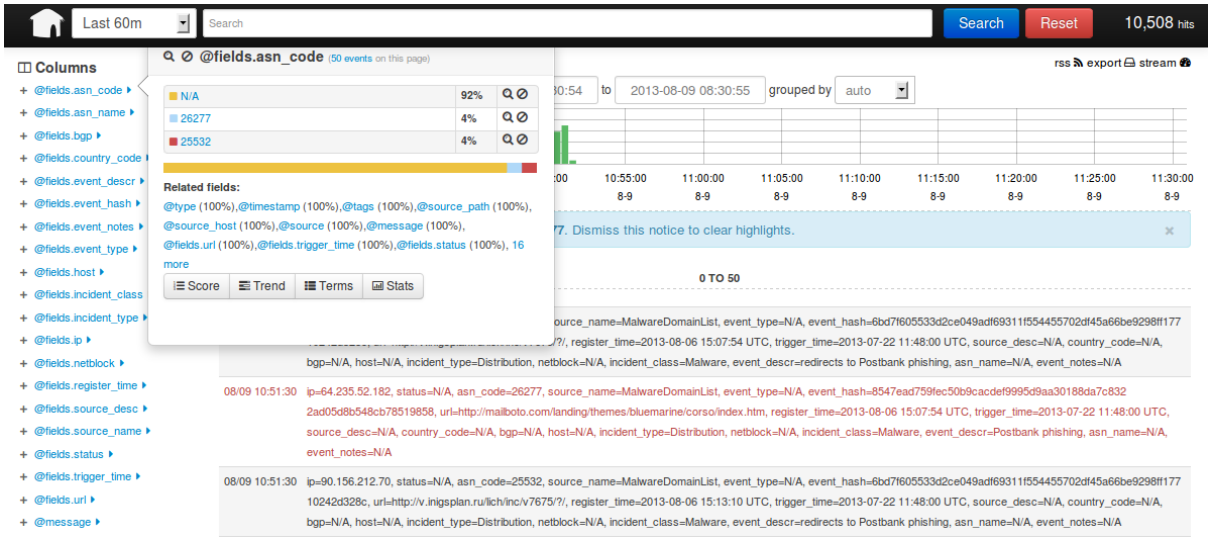

Figure 21: Kibana filter screenshot 2

term will change the way all data is displayed in the main section (shown in the screenshot):
In this case only the content of the changed data field is shown. Clicking the field name itself will open a popup window with several options:

You will be given a shortlist of the data available in the displayed data (only the 50 lines which are shown, not all data sets available). Clicking one of these lines will highlight the matching data sets (red lines in the screenshot). Clicking the magnifier symbol will execute a search using this term (see screenshot below).

At the bottom of the popup additional features are available:

- Score: Listing of all the instances of this category ordered by count of appearance.
- Trend: Listing ordered by relative change of appearance.

- Stats: Statistics are not available for the data in this exercise.

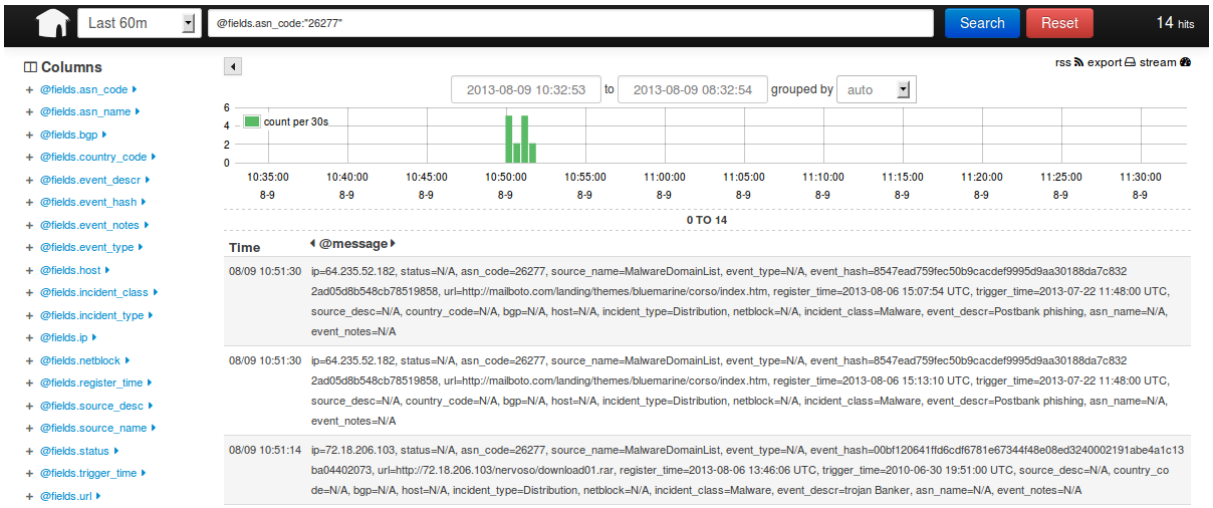

Figure 12: Kibana filter screenshot 3

This is also an example for the use of the text field. You can either enter a plain term ('example') and execute a full text search for it or address a single field (or a combined set of fields, use 'AND' and 'OR') from the list on the left side ('@ and search (filter) for specific content in this data set. To address a data field use the 'at sign' as a prefix. List items starting with 'fields.' are created from the variable names in the data itself. Other items are standard set provided by Kibana.

Examples:

@fields.asn_code:"26711"

@fields.asn_code:"26711" AND @fields.incident_class:"Phish"

A more detailed explanation of the query syntax can be found on the Apache Lucene site.[30]

5.5 Task 3: Cybercrime scenarios

For each scenario the trainer (or the students) executes the netcat command which injects the data to be analysed into the current data flow generated by the AbuseHelper feeds. The data will be forged to appear as being generated by an existing AbuseHelper bot and represent the corresponding infrastructure component. It will be sanitised to fit the categories defined in logstash and used in Kibana to dissect the information.

There will be different attributes of the data which can be used by the students to identify and analyse the incidents. There is data linked to the network (IP/ASN) of the company. There are hints on targets similar to the company and there are logs generated by the company's systems (Webserver/IDS).

5.5.1 Phishing

A phishing campaign targets the company's customers to acquire login data to the administrative access of the shopping solutions.

Data:

- Phishing alerts containing sites with faked example.com domain names;

[30] *Apache Lucene Query Syntax* – https://lucene.apache.org/core/3_5_0/queryparsersyntax.html

- Logs containing successful logins from unusual source addresses.
- Questions to be answered:
- The customers are based in the EU and logins from networks outside Europe are very unusual. Please check the logins and write down any anomalies.
- Check the incident data for hints on the cause of the compromises.
- For both parts develop and document the next steps during the incident response process.

The data file resides in the folder /usr/share/trainee/28_VCT/add/event_, change your working directory to it and execute the command ***'netcat localhost 5544 < event-log.csv'***. The file consists mostly of data not related to the event ('aka background noise') and a few lines of crafted incident-related information injected into it.

Filter for '***@fields.url:"example.com"***' to find data related to fraudulent login sites.

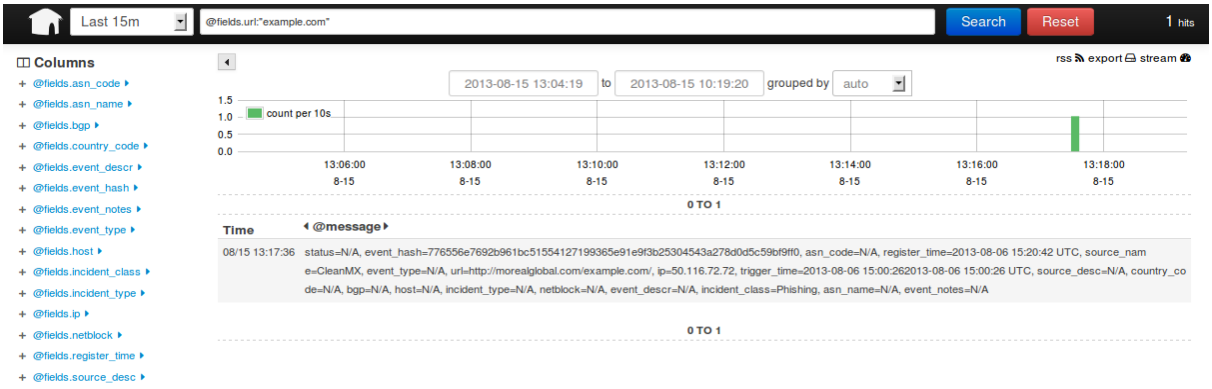

Figure 33: Kibana phishing filter screenshot

Filter for '***@fields.source_name:"tailbot" AND NOT @fields.country:"EU"***' to identify logins unlikely to be legitimate according to the company's customer base.

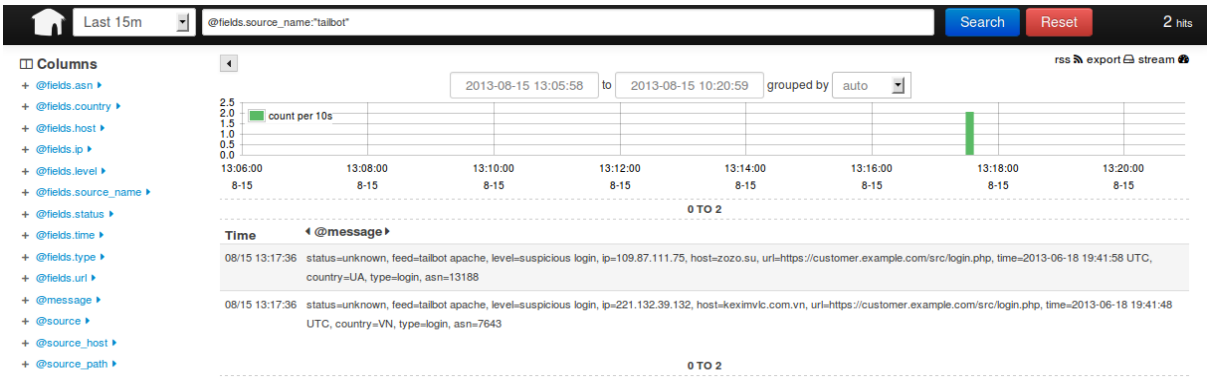

Figure 44: Kibana login filter screenshot

5.5.2 Trojan

With the help of acquired login credentials attackers are placing malware on the hosted sites and compromise client systems through drive-by downloads.

Data:

- Malware alerts regarding the network range / ASN owned by the example company.
- Questions to be answered:
- The data contains evidence for the next steps the attackers have taken after the initial compromise. Identify the incidents and document the information and the necessary steps to contain and respond to them.

The data file resides in the folder /usr/share/trainer/28_VCT/add/event_2, change your working directory to it and execute the command ***'netcat localhost 5544 < event-log.csv'***. The file consists mostly of data not related to the event (''aka background noise') and a few lines of crafted incident-related information injected into it.

Filter for '***@fields.asn_code:"26711" AND @fields.incident_class:"Malware"***' to find malware distribution alerts related to the company's network.

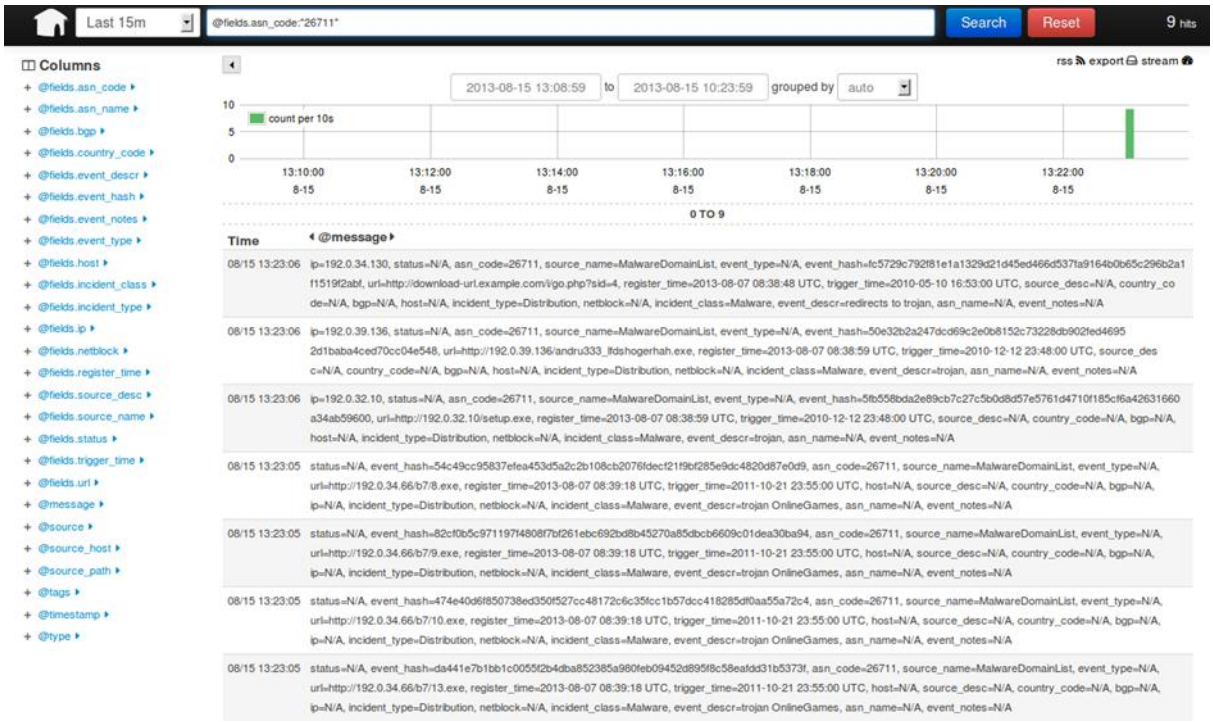

Figure 15: Kibana malware filter screenshot

5.5.3 Extortion

Attackers started a DDoS against the hosting infrastructure and threaten to take the company down unless it pays a ransom.

Data:

- Webserver Logs
- IDS Alerts (ModSecurity)
- Questions to be answered:

- Customers are complaining about issues with webservers. Check the data for possible causes and document them with technical detail.

The data file resides in the folder /usr/share/trainer/28_VCT/add/event_3, change your working directory to it and execute the command '***netcat localhost 5544 < event-log.csv***'. The file consists mostly of data not related to the event ('aka background noise') and a few lines of crafted incident-related information injected into it. The incident data has been generated by attacking an Apache Webserver with the DoS tool Tor's Hammer.[31]

Filter for '***@fields.source_name:"apache" AND @fields.log_message:"post"***' to identify the log lines related to the attack.

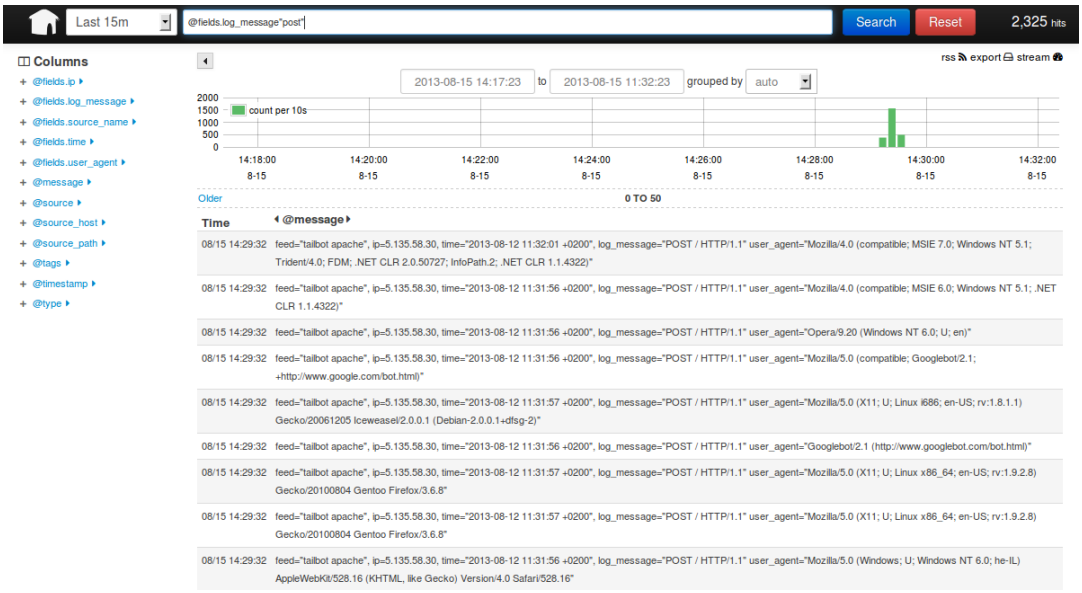

Figure 56: Kibana DoS filter screenshot 1

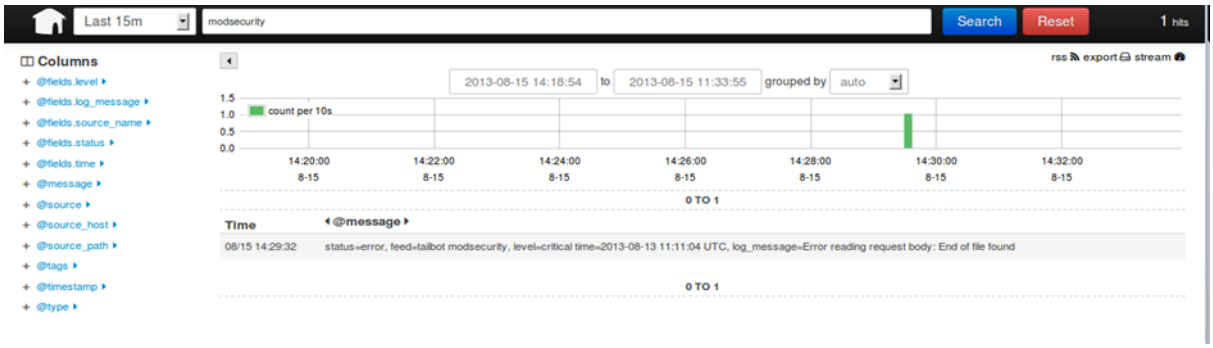

Figure 17: Kibana DoS filter screenshot 2

[31] http://packetstormsecurity.com/files/98831/Tors-Hammer-Slow-POST-Denial-Of-Service-Testing-Tool.html

6 Summary

During the summary each part of the exercise should be reviewed and evaluated. Start with the results of Task 3 and let each team present what they found in the data and explain how they found it. Afterwards the students should discuss the different approaches.

In the second part of the summary the students should review Kibana as a visualisation tool for incident response data. Let them compare it to other implementations (either from the introduction or from individual knowledge). Make notes to be able to summarise the results from this discussion in the end.

Finally, the discussion focuses on the general approach of visualising information. Elaborate the advantages and disadvantages of visualisation.

7 References

1. The Honeynet Project – Know Your Tools use Picviz to find attacks, 2009 (http://www.honeynet.org/files/KYT-Picviz_v1_0.pdf)
2. logstash – open source log management (http://logstash.net/)
3. Kibana – Make sense of a mountain of logs (http://kibana.org/)
4. Elasticsearch – Real Time Data and Analysis (http://www.elasticsearch.net/)
5. certpt – GenericEvent-AbuseHelper (https://bitbucket.org/certpt/genericevent-abusehelper/src)
6. certpt – LogCollector-AbuseHelper-Extension (https://bitbucket.org/certpt/logcollector-abusehelper-extension/overview)
7. Wikipedia – Information visualization (https://en.wikipedia.org/wiki/Information_visualization)
8. Apache Lucene – Query Parser Syntax (https://lucene.apache.org/core/3_5_0/queryparsersyntax.html)
9. CERT.at – ProcDOT (http://cert.at/downloads/software/procdot_en.html)
10. Packet Storm – Tor's Hammer (http://packetstormsecurity.com/files/98831/Tors-Hammer-Slow-POST-Denial-Of-Service-Testing-Tool.html)
11. Wikipedia – Netcat (http://en.wikipedia.org/wiki/Netcat)
12. ENISA – Exercise Material (http://www.enisa.europa.eu/activities/cert/support/exercise)
13. AbuseHelper – (http://www.abusehelper.be/)
14. Wikipedia – Graph (abstract data type) (https://en.wikipedia.org/wiki/Graph_%28data_structure%29)
15. Wikipedia – Parallel coordinates (https://en.wikipedia.org/wiki/Parallel_coordinates)
16. Wikipedia – Heatmap (https://en.wikipedia.org/wiki/Heatmap)
17. Team Cymru – Internet Malicious Activity Maps (https://team-cymru.org/Monitoring/Malevolence/maps.html)

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