Bitdefender®

Hiding the network behind the network

Botnet proxy business model

Alexandru Maximciuc **Cristina Vatamanu**

Razvan Benchea

Overview

- General information
- Proxy Level 1
- Central DNS SERVER
- Abuse reports
- Statistics
- Conclusion

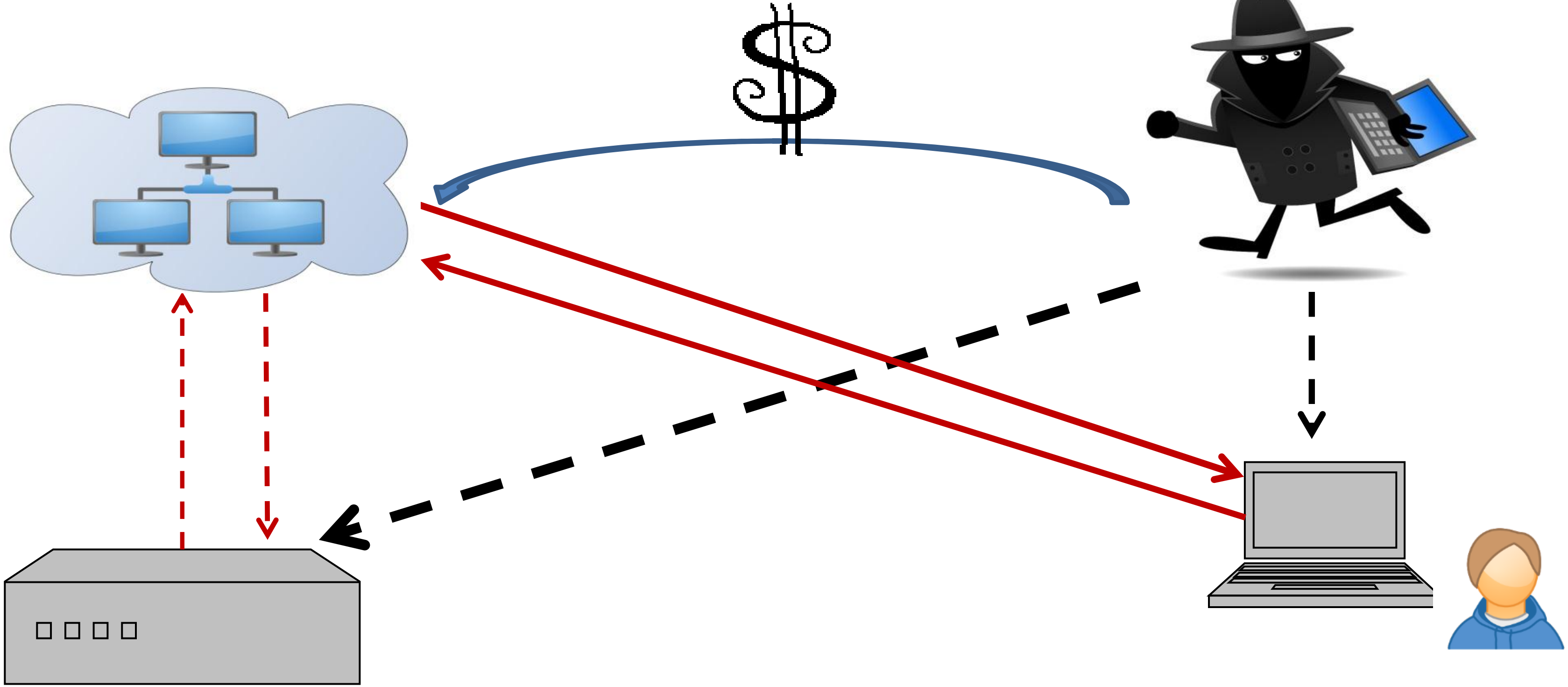
General information

- Botnets
  - infects users’ computers
  - contacts C&C and waits for commands
  - when it receives them the payload is executed
- Typical responses from AV companies: blacklist and takedowns

General information

- Main interest → to ensure a long functionality and anonimization for the C&C
- Evolution: DGAs (not enough)
- Strong demand for a solution
- Therefore it was inevitable not to see an offer with specialized systems which can ensure a good anonimization

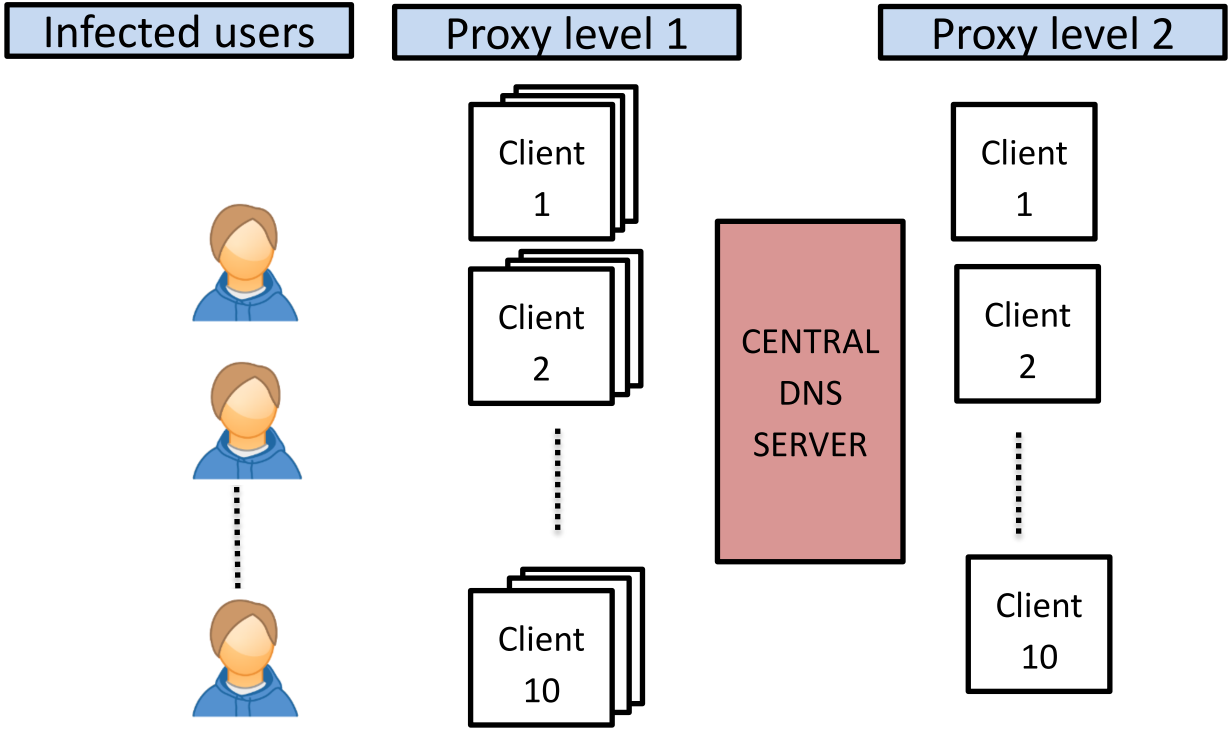
General information



General infrastructure

- Two levels of proxy protecting C&C servers
- A central DNS server handling UDP and HTTP traffic
- Architecture flexible to rapid changes
- Serving different kinds of malware families

General infrastructure



Copyright@bitdefender 2014 / www.bitdefender.com

Proxy level 1

- Responsible for redirecting
  - the UDP traffic (on port 53)
  - the HTTP traffic (usually on port 80)

Proxy level 1. UDP redirection

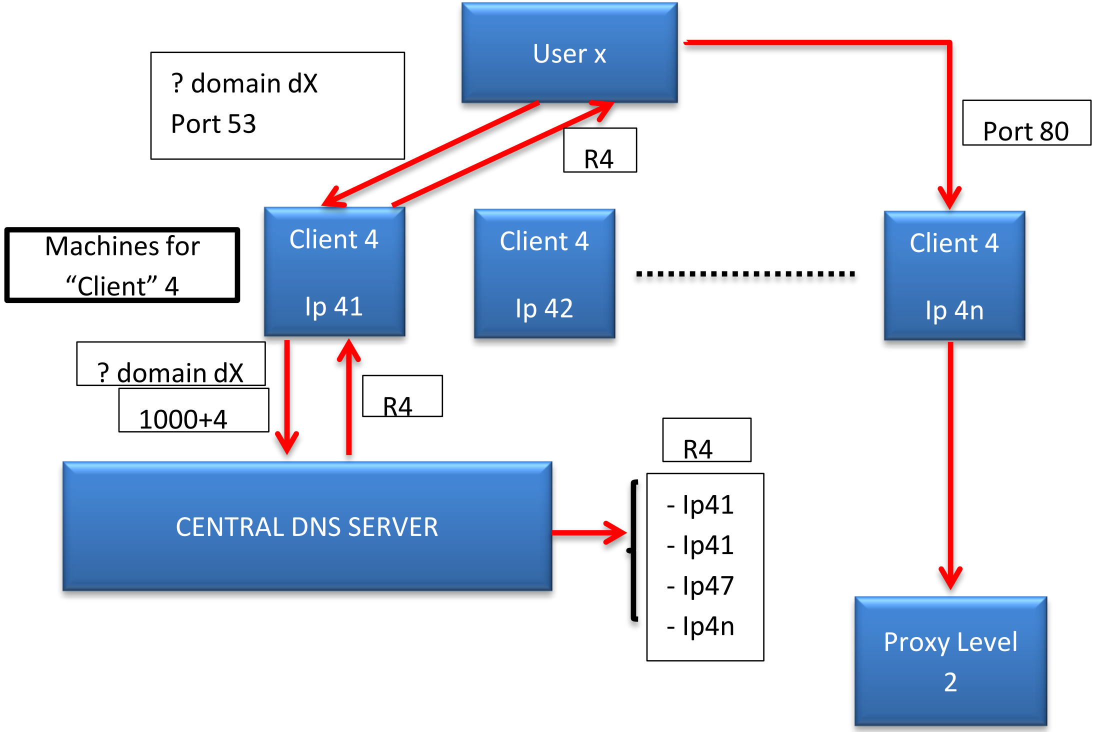
- First level proxy machines are set as authoritative name servers for different domain names; any DNS resolution request arrives here
- All the traffic received on port 53 is redirected to a central DNS SERVER
- The port used for this redirection is 1000 + client_id
- This server responds with 4 alive IPs, randomly chosen from the list of IP addresses allocated to the current client

Proxy level 1. HTTP redirection

- The victim’s computer choses one of these IP address and sends a HTTP request to the machine corresponding to it.

- This machine will redirect this request on a machine from the second level proxy, usually on port 80.

Proxy level 1



Proxy level 1. Redirection service

- Major component: an encrypted binary file (elf) named *map:*
  - Self-update functionality
  - Update for service.xml, the service responsible with traffic redirection.
- The structure of the service.xml

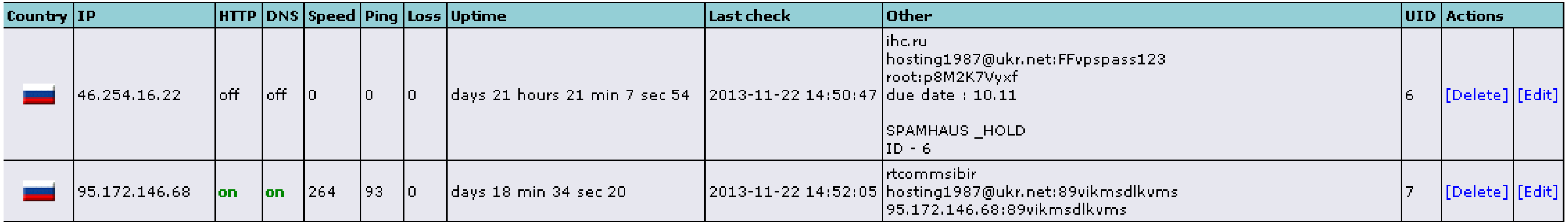
```
<?xml version=”1.0” encoding “UTF-8” ?>
<tunnel>
      <tunnel from=’http’ from_port=’80’ to=’http’ to_port=’80’> IP_PROXY_LEVEL_2
      </tunnel>
       <tunnel from=’udp’ from_port=’53’ to=’udp’ to_port=’1000+client_id’>IP_DNS_SERVER
       </tunnel>
</tunnel>
```

Central DNS Server

- Three main activities:
  - resolves DNS queries
  - serves updates for service.xml
  - represents the management interface for all the “clients”
- During our investigation the DNS SERVER was moved from one machine to another
- Collection of php scripts was analyzed, divided in three main categories: admin, checker and system

Central DNS Server

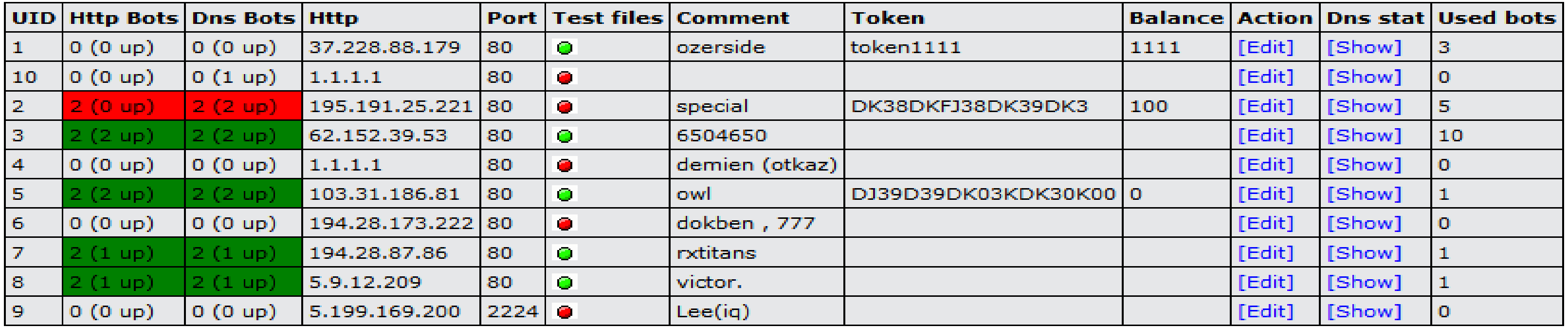
- Admin. *index.php*
  - Received commands:
    - *del* - deletes IPs from *servers* table
    - *edit* - edits IPs from *servers* table
    - *<without parameters>* - displays information



Country	IP	HTTP	DNS	Speed	Ping	Loss	Uptime	Last check	Other	UID	Actions	
🇷🇺	46.254.16.22	off	off	0	0	0	days 21 hours 21 min 7 sec 54	2013-11-22 14:50:47	ihc.ru hosting1987@ukr.net:FFvpspass123 root:p8M2K7Vyxf due date : 10.11 SPAMHAUS _HOLD ID - 6	6	[Delete]	[Edit]
🇷🇺	95.172.146.68	on	on	264	93	0	days 18 min 34 sec 20	2013-11-22 14:52:05	rtcommsibir hosting1987@ukr.net:89vikmsdlkvms 95.172.146.68:89vikmsdlkvms	7	[Delete]	[Edit]

Central DNS Server

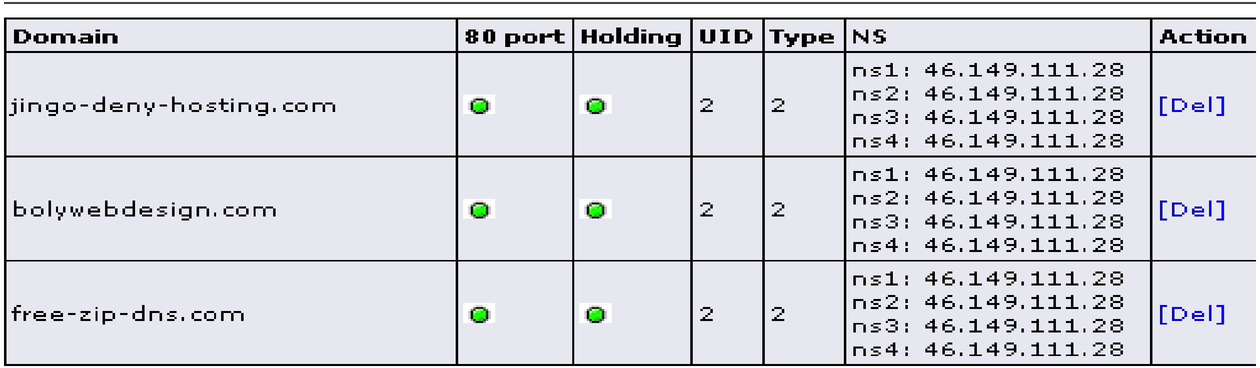
- Admin. *users.php*
  - Received commands:
    - *add*– parameters as *ip, port, comment* are saved in the client's corresponding file
    - *edit* – previously mentioned parameters are shown on the web page and allows their actualization
    - *<without parameters>* - displays information



UID	Http Bots	Dns Bots	Http	Port	Test files	Comment	Token	Balance	Action	Dns stat	Used bots
1	0 (0 up)	0 (0 up)	37.228.88.179	80	🟢	ozerside	token1111	1111	[Edit]	[Show]	3
10	0 (0 up)	0 (1 up)	1.1.1.1	80	🔴				[Edit]	[Show]	0
2	2 (0 up)	2 (2 up)	195.191.25.221	80	🔴	special	DK38DKFJ38DK39DK3	100	[Edit]	[Show]	5
3	2 (2 up)	2 (2 up)	62.152.39.53	80	🟢	6504650			[Edit]	[Show]	10
4	0 (0 up)	0 (0 up)	1.1.1.1	80	🔴	demien (otkaz)			[Edit]	[Show]	0
5	2 (2 up)	2 (2 up)	103.31.186.81	80	🟢	owl	DJ39D39DK03KDK30K00	0	[Edit]	[Show]	1
6	0 (0 up)	0 (0 up)	194.28.173.222	80	🔴	dokben , 777			[Edit]	[Show]	0
7	2 (1 up)	2 (1 up)	194.28.87.86	80	🟢	rxtitans			[Edit]	[Show]	1
8	2 (1 up)	2 (1 up)	5.9.12.209	80	🟢	victor.			[Edit]	[Show]	1
9	0 (0 up)	0 (0 up)	5.199.169.200	2224	🔴	Lee(iq)			[Edit]	[Show]	0

Central DNS Server

- Admin. *domains.php*
  - Received commands:
    - *del* - deletes the domain from the *domains* table
    - *add* – registers domains through ***cnobin.com*** and inserts the data in the *domains* table (domain, uid, ns1, ns2, ns3, ns4)
    - *<without parameters>* - displays information



Domain	80 port	Holding	UID	Type	NS	Action
jingo-deny-hosting.com	●	●	2	2	ns1: 46.149.111.28 ns2: 46.149.111.28 ns3: 46.149.111.28 ns4: 46.149.111.28	[Del]
bolywebdesign.com	●	●	2	2	ns1: 46.149.111.28 ns2: 46.149.111.28 ns3: 46.149.111.28 ns4: 46.149.111.28	[Del]
free-zip-dns.com	●	●	2	2	ns1: 46.149.111.28 ns2: 46.149.111.28 ns3: 46.149.111.28 ns4: 46.149.111.28	[Del]

Central DNS Server

- Checker. *checker.php*
  - Sets information in the *servers* table:
    - *column* ***http*** - [**on|off**] if it receives a valid answer from the servers it sets the column to *on,* otherwise to *off*
    - *column* ***dns*** - [**on|off**] if it receives a valid answer from the servers it sets the column to *on,* otherwise to *off*
    - *column* ***http_good*** - [**on|off**] if certain conditions are met, the column is set to *on,* otherwise to *off*
    - *column* ***dns_good*** - [**on|off**] if certain conditions are met, the column is set to *on,* otherwise to *off*

Central DNS Server

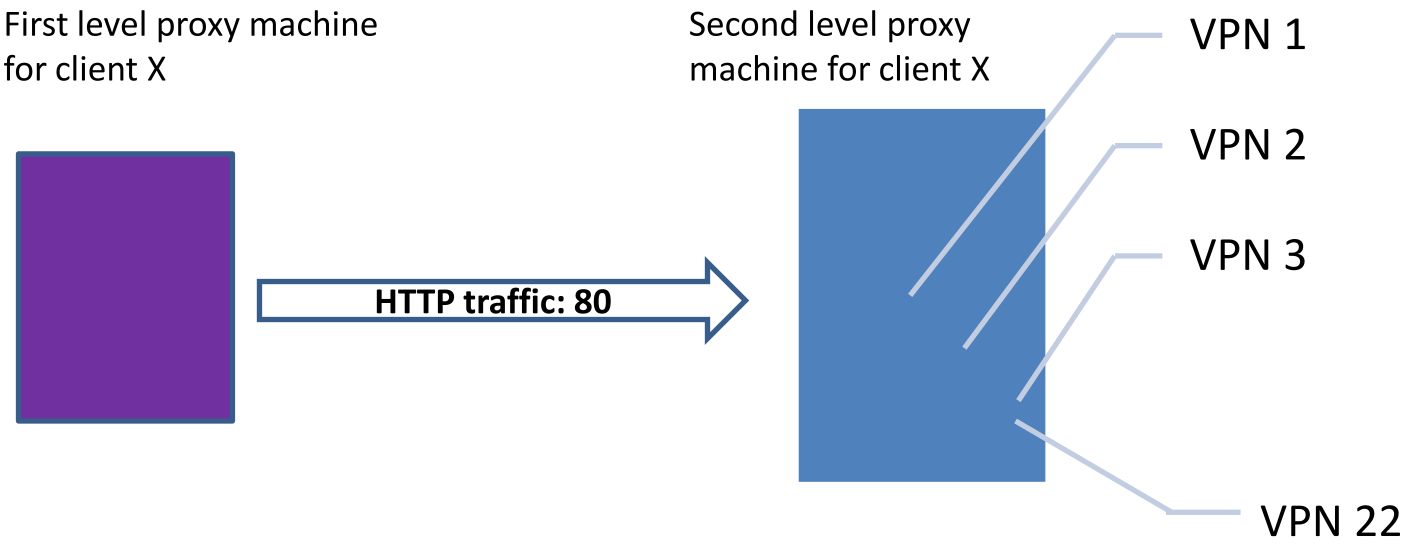
- System
  - scripts for RC4 encryption and decryption
  - config files
  - scripts that delete from the database the servers that have an “expired” LastCall
  - template for “service.xml”

Config.ini

```
<tunnels>
  <tunnel from ‘http’ from_port=‘80’ to=‘http’ to_port=%http_port%><%http_ip>
  </tunnel>
  <tunnel from ‘udp’ from_port=‘53’ to=‘udp’ to_port=%dns_port%><%dns_ip>
  </tunnel>
</tunnels>
```

Proxy level 2

- Network anonymisation through tunneling technique (frontend, backend, node, vdcr roles)
- A variable number of opened VPNs


First level proxy machine for client X
Second level proxy machine for client X
HTTP traffic: 80
VPN 1
VPN 2
VPN 3
VPN 22

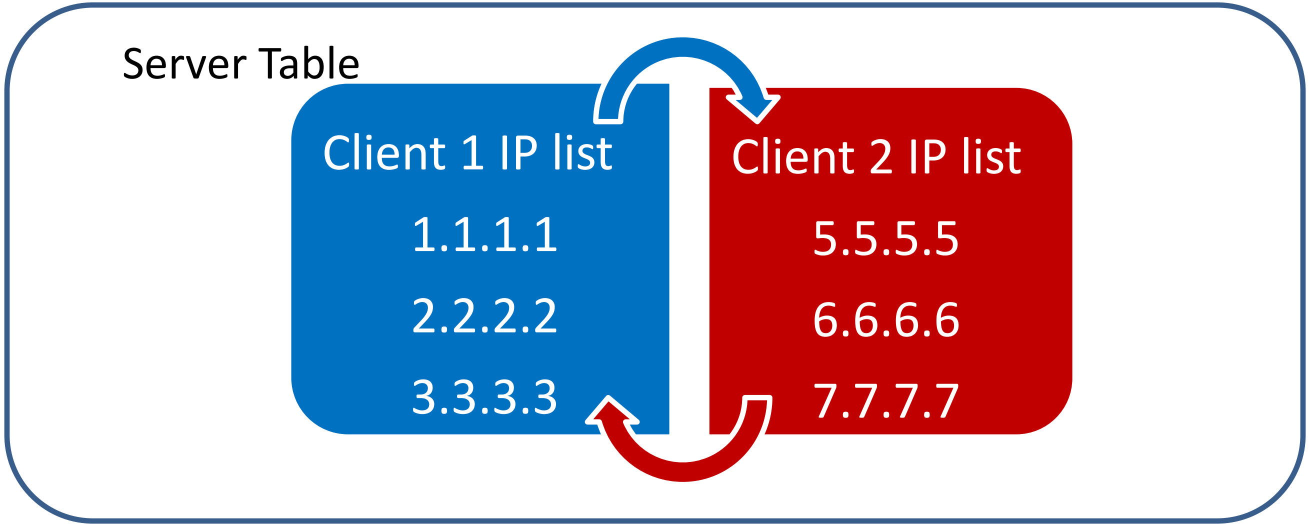
Abuse reports

- This complex network architecture proves to be very effective in case of abuse reports.

- We submitted two types of abuse reports and every time the network recovered very quickly.

Abuse reports at first level proxy

- Solution
  - switch between clients IP lists

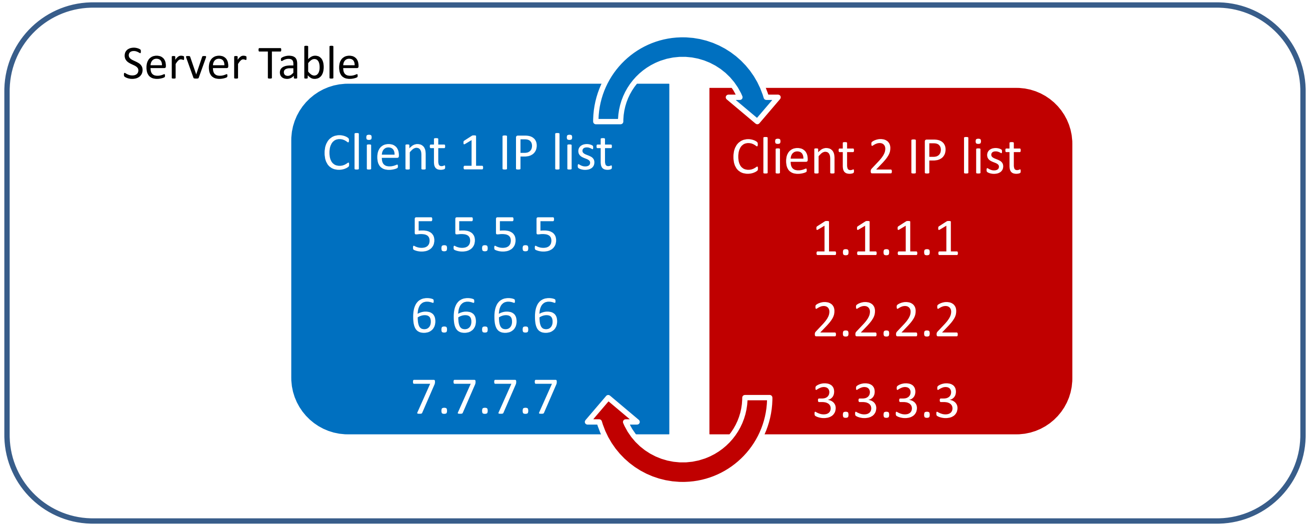


Server Table

Client 1 IP list	Client 2 IP list
1.1.1.1	5.5.5.5
2.2.2.2	6.6.6.6
3.3.3.3	7.7.7.7

Abuse reports at first level proxy

- Solution
  - switch between clients IP lists

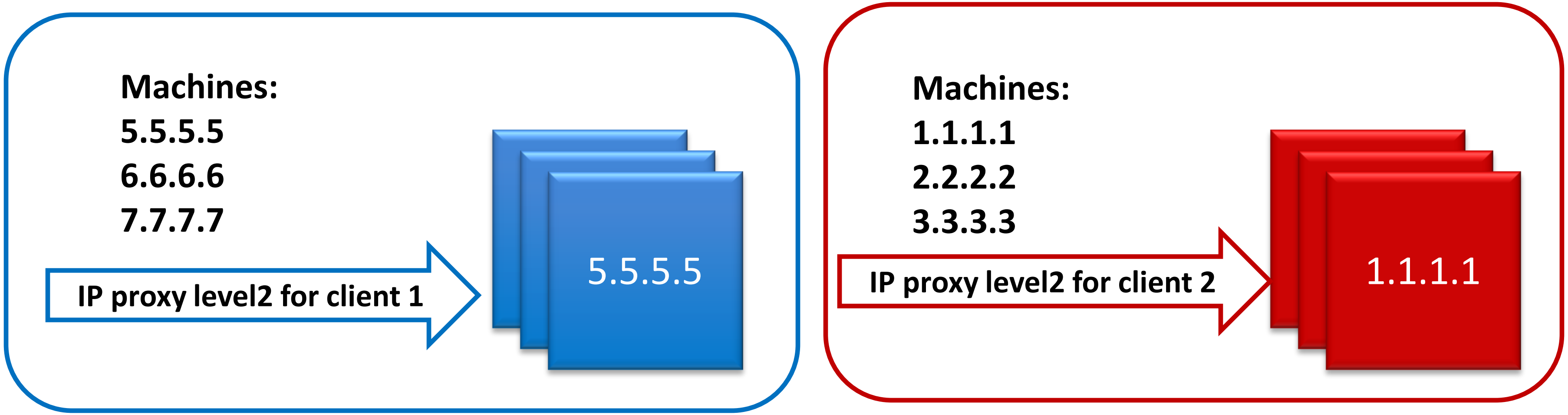


Server Table

Client 1 IP list	Client 2 IP list
5.5.5.5	1.1.1.1
6.6.6.6	2.2.2.2
7.7.7.7	3.3.3.3

Abuse reports at first level proxy

- Solution
  - an update for service.xml file to correct HTTP traffic redirection



Machines:
5.5.5.5
6.6.6.6
7.7.7.7

Machines:
1.1.1.1
2.2.2.2
3.3.3.3

Abuse reports at second level proxy

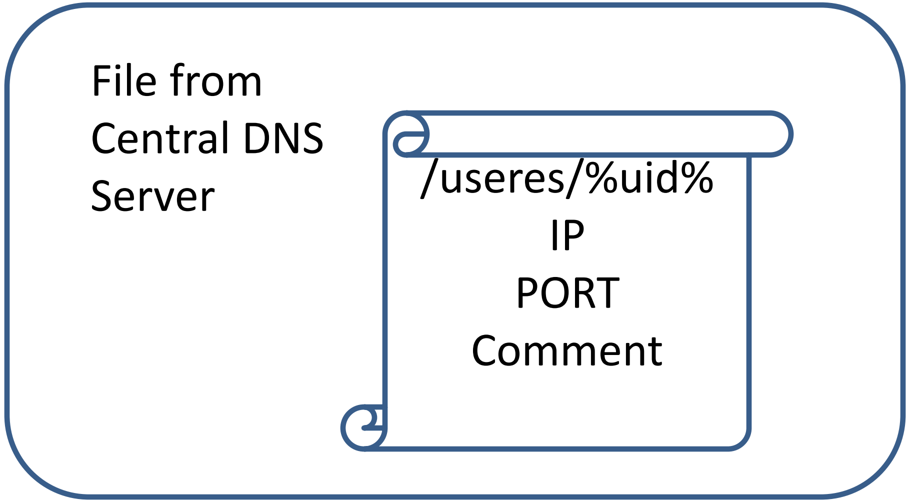
- The machine is stopped

- In approximately 3-4 hours, a new IP appears in the system

- In less than 24 hours, the malware is back in business

Abuse reports at second level proxy

- Solution
  - Replaced the old IP with the new one in the clients corresponding file from the Central DNS Server

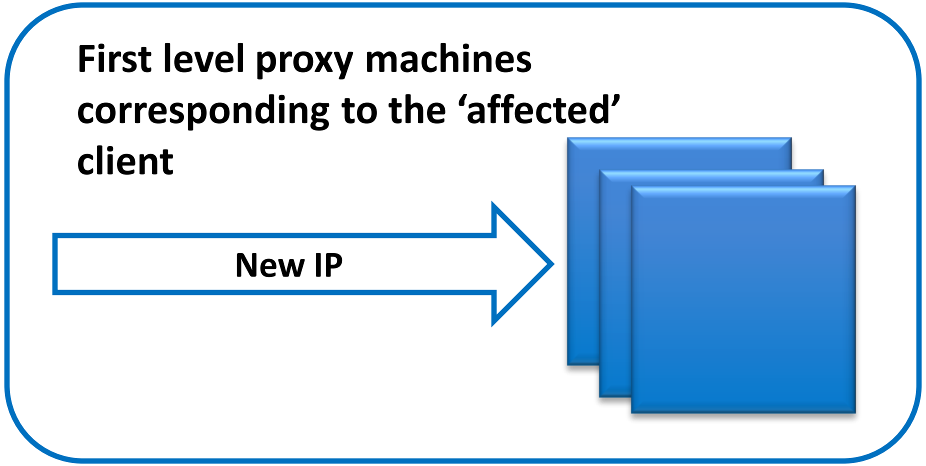


File from Central DNS Server

/useres/%uid%
IP
PORT
Comment

Abuse reports at second level proxy

- Solution
  - Updates service.xml on all corresponding first level machines


First level proxy machines corresponding to the 'affected' client
New IP

Cryptolocker Story

- It was the client with the _ID = 2, named *“special”*

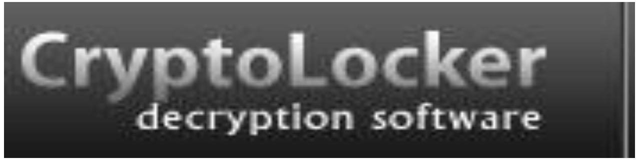
- At the moment of takedown (**2-nd of June**) the registered domain names did not resolve to the first level proxy IPs

- On **10-th of June** first attempt to recover (one new IP for first proxy level and one for second proxy level)

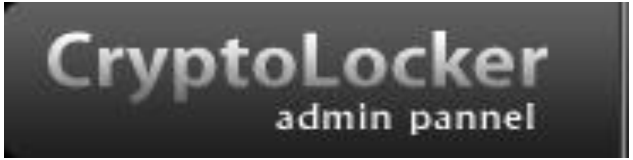
- The attempt was unsuccessful, the IPs were removed from the network just a few minutes later

Cryptolocker Story

- On **5-th**, **6-th** and **8-th of August** they added in the system new IPs for first proxy level and second proxy level
- None of them responded as a valid Cryptolocker IP but on */img* it was an open directory revealing

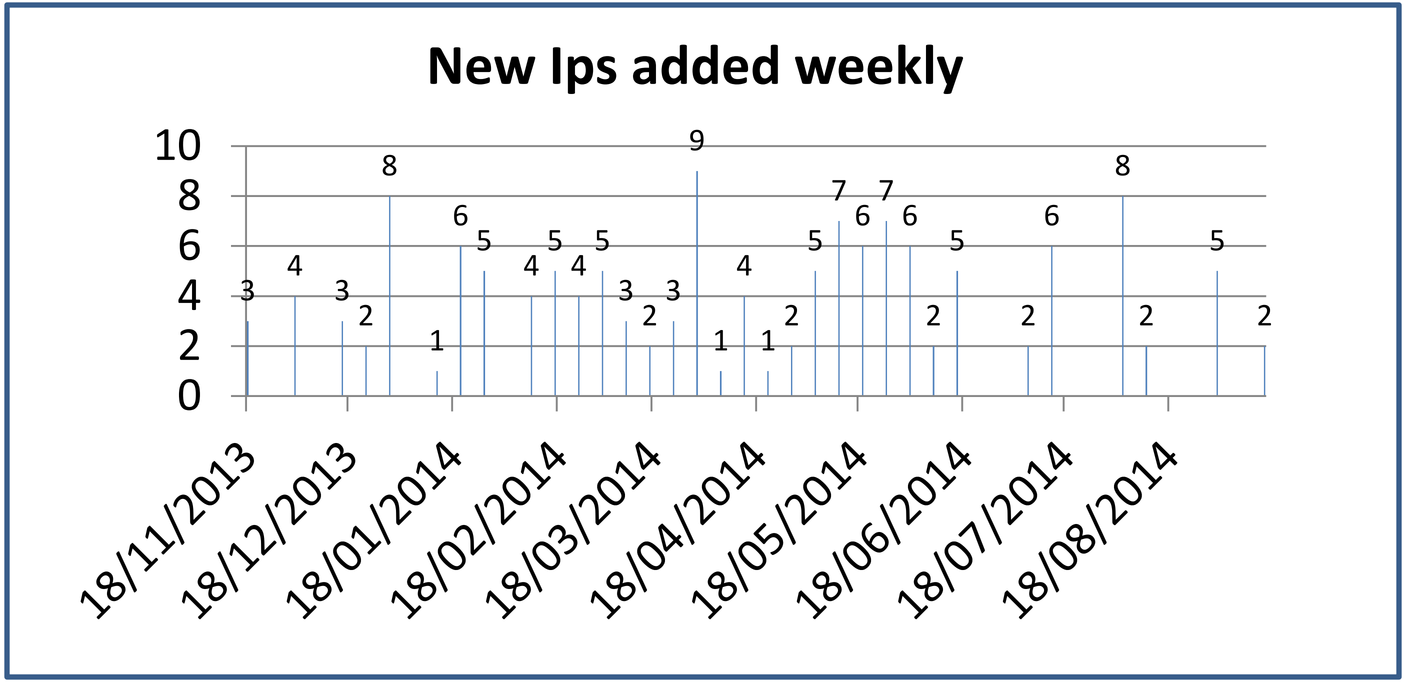


CryptoLocker
decryption software

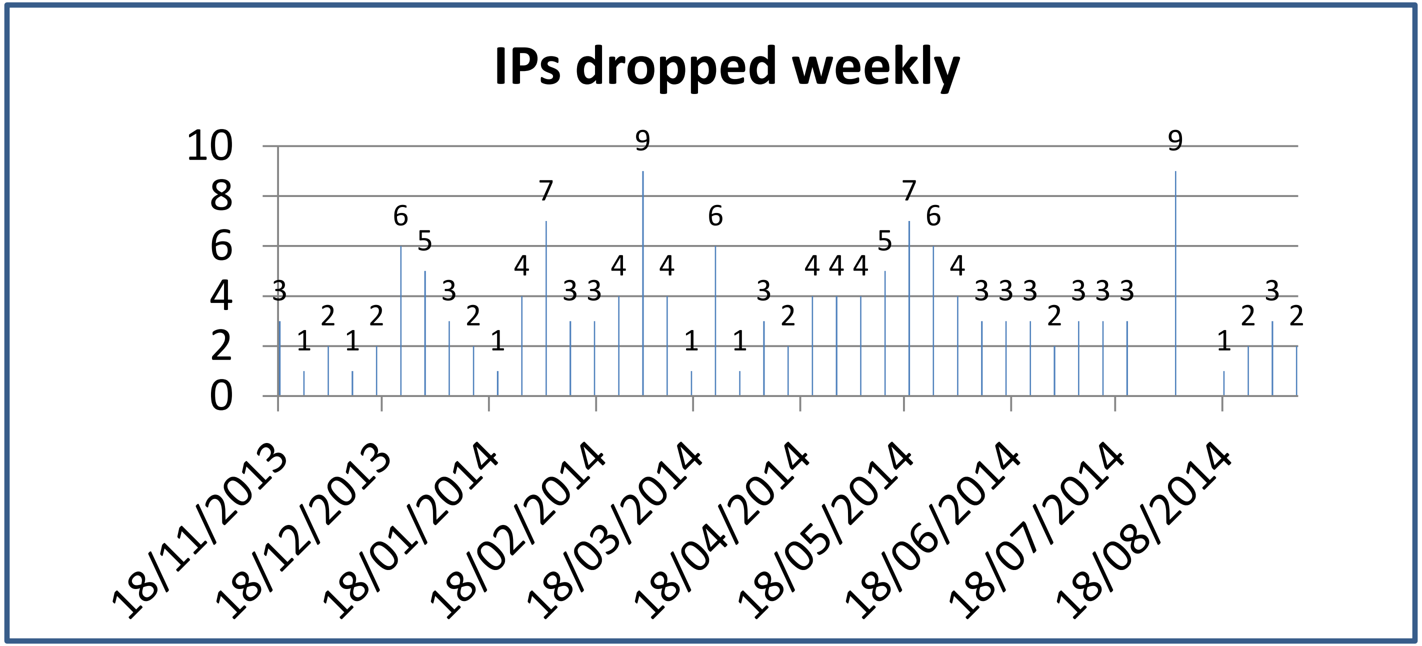


CryptoLocker
admin pannel

Statistics

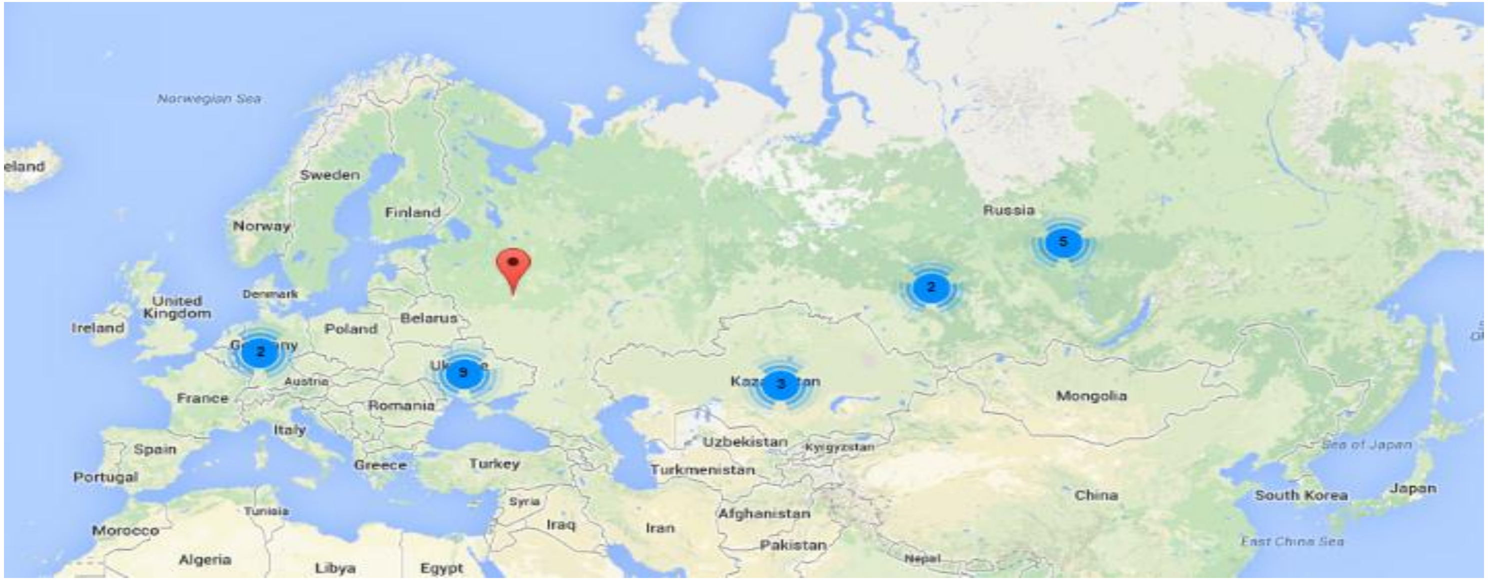

New Ips added weekly
10
8
6
4
2
0
3
4
3
2
8
1
6
5
4
5
4
5
3
2
3
9
1
4
1
2
5
7
6
7
6
2
5
2
6
8
2
5
2
18/11/2013
18/12/2013
18/01/2014
18/02/2014
18/03/2014
18/04/2014
18/05/2014
18/06/2014
18/07/2014
18/08/2014

Statistics



IPs dropped weekly

Statistics. September 2014



Conclusions

- The network proved to be very resistant to abuse reports
- The time needed to recover is very short
- It represent a good solution for malware creators who want to hide their C&C
- The network resisted on the market for quite a while → we expect similar mechanism to appear on the botnet market

Q&A