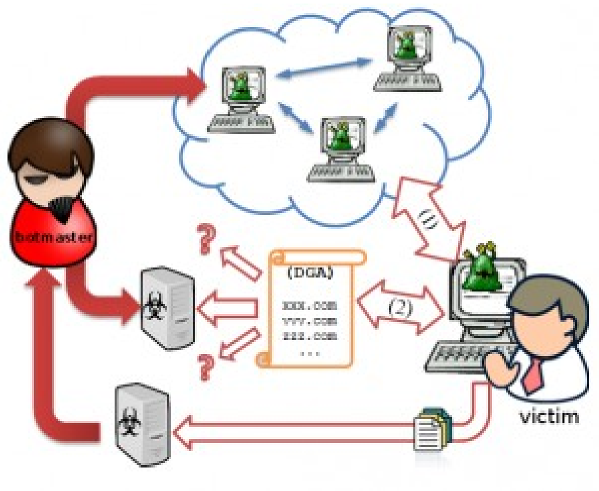
ZeuS – P2P+DGA variant – mapping out and understanding the threat

In the autumn of 2011 we observed new malware infections, which looked similar to Zeus. Subsequent analysis of the malicious software mechanism start up, the process of hiding and storing of configuration indeed verified that it was ZeuS. However, monitoring of infected machines failed to uncover the characteristic communicatation with a C&C. After closer examination it appeared that the sample was probably a new version based on the source code of ZeuS that wasaccidentally made public.



In the new version of the Trojan, the authors focus on eliminating the weakest link – a centralized system of information distribution.
Previous versions of Zeus were based on one (or few) predefined addresses which were used for botnet management. This allowed for relatively easy tracking and blocking of servers, thus rendering the botnet useless. However, the analysed variant of the Trojan used two new channels of communication to receive orders (figure on right):

1. Communication in a peer-to-peer network

2. Domain names Generation Mechanism

This variant has been analyzed to some extent by other researchers before – there is information on the web on the new variant of Zeus (eg abuse.ch), however – based on our knowledge – previous research has focused on registering and monitoring traffic to Zeus domains. **In our work we focus on understanding the P2P network communication mechanisms, mapping out the network, and monitoring the exchange of information in this particular network.**

Information sharing over the Zeus-peer-to-peer (ZP2P) network

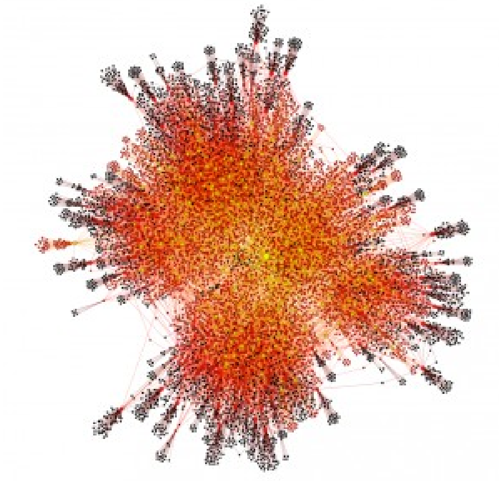


fig.1: Visualization of network ZP2P(10 000 nodes)

In the case of a model based on central (one – or many) point of management, it is possible to identify machines used for command & control. The new mechanism for distributing information is based on the direct exchange of data among infected computers – a model of communication based on peer-to-peer networks. The fact that there is no central management node in this model makes it much harder to find which computer is used to distribute new orders – and the blocking of the information exchange channel is virtually impossible. This is well illustrated by the graph in Figure 1. You can see that the presented network does not contain any central point (single or multiple), and the connections between computers are random.

How does the ZP2P network operate?

The network most likely is based on Kademlia protocol standard. A single computer (node) in ZP2P network is identified by a unique identifier UID – which is generated during the first run of malicious software. Each of the computers belonging to the ZP2P has a “table of neighbors” stored in memory. This array contains a list of about 30 neighboring nodes in the ZP2P network – their UID, IP address and UDP port number. This list is used to exchange binary data and information.

In the ZP2P network, we can distinguish two types of communication:

- The exchange of administrative information (using UDP):
  - (QV) Exchange of information about version of the configuration files
  - (QN) Exchange of information about the nodes in the “table of neighbors”
- The exchange of binary data (using the TCP protocol)
  - Distribution of new configuration files

Charts 2 and 3 show the distribution of port numbers used for network communication by ZP2P on the network we mapped.

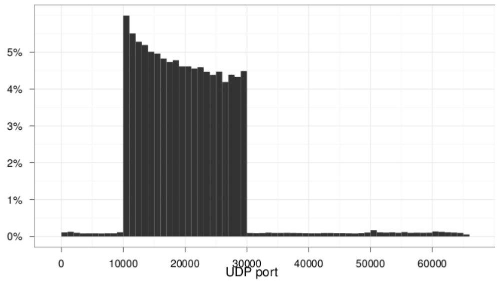


chart.2: Distribution of UDP port numbers in the ZP2P network (800 000 samples)

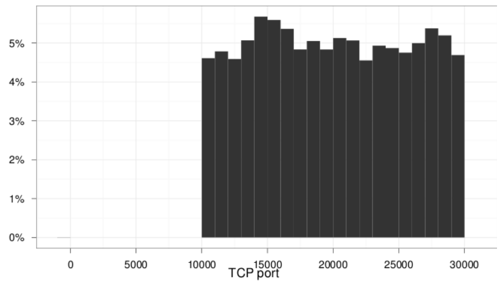


chart.3: Distribution of TCP port numbers in the ZP2P network (100 000 samples)

If the case of message type QN, only 10 records from the "table of neighbors" are sent back to client. This type of communication is dedicated for upgrading the local "table of neighbors". After the query type QN, the bot saves information about nodes whose UID is similar to the bot UID (XOR metric) on a local computer.

Messages of type QV are used for checking and the propagation of information about new versions of configuration files. If the node that performed the query QV has an older version of the configuration than the version given in response to this query – the bot performs a TCP connection to a remote computer asking for a newer version of the configuration.

Charts 4 and 5 show the distribution of the number of children and the number of parents in the ZP2P network. The data comes from the analysis of responses to QN queries during a 3 week period. The number of children (the number of entries in "table of neighbors") may exceed the value 30, because (as already described), the table may be frequently updated.

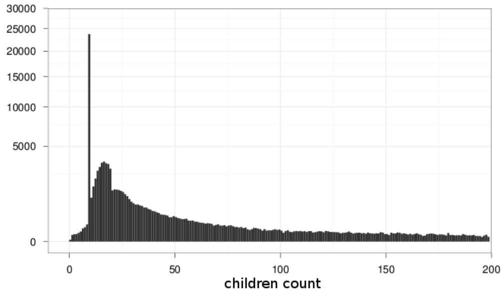


chart.4: Distribution of number of children

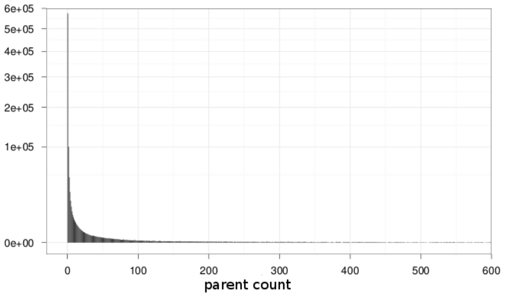
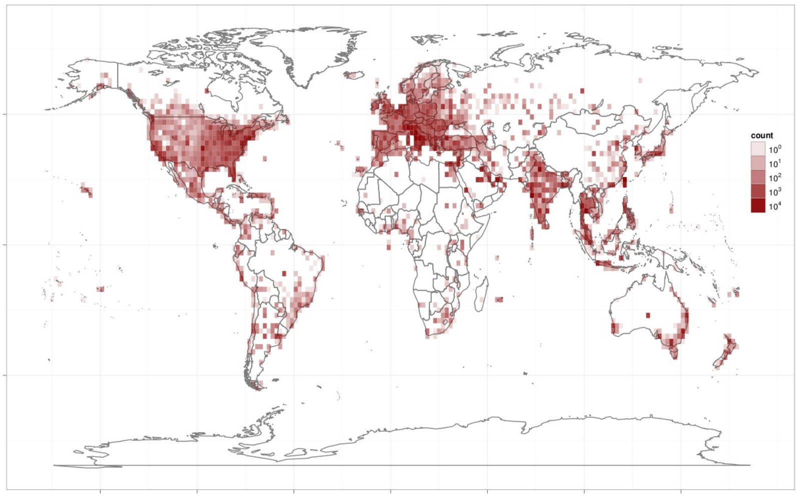


chart.5: Distribution of number of parents

ZP2P Network Monitoring

In the CERT Polska laboratory, we manage to do a mapping of a ZP2P network by monitoring the response to type QN queries. Collected IP addresses are plotted on a map:

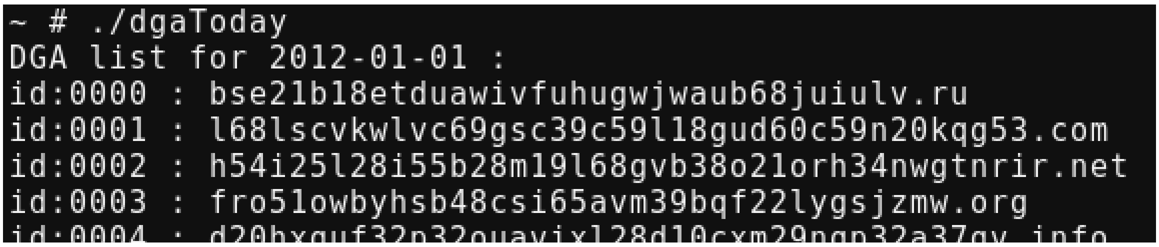


Map of density of number of computers infected with new variant of the Trojan

The DGA mechanism

If the communication mechanism of the ZP2P network will be blocked (eg by blocking the appropriate TCP and UDP ports on a firewall) – the bot will automatically switch to the backup communication channel – DGA. DGA mechanism is another feature implemented in the new version of the Trojan. It significantly impacts the difficulty of finding and the cutting-off of the miscreants behind the botnet.

The DGA allows for the generation of a long list of domain names based on specified parameters, and subsequently of attempts to communicate with each of the generated domains. DGA mechanism parameters are hidden inside the Trojan code – and are known only to the botmaster. The botmaster can manually generate such a list, select one position, register the selected domain, and wait for connection attempts from infected machines.



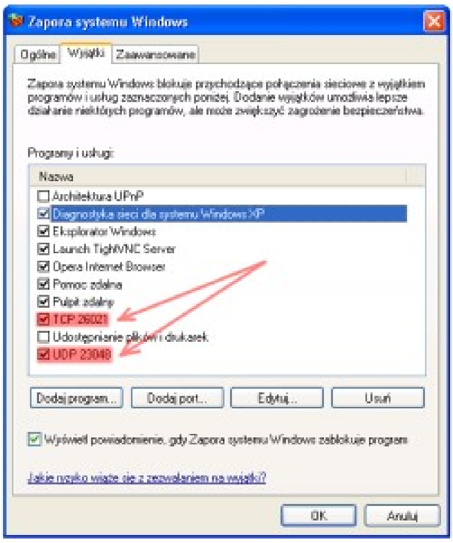
```
~ # ./dgaToday
DGA list for 2012-01-01 :
id:0000 : bse21b18etduawivfuhugwjwaub68juiulv.ru
id:0001 : l68lscvkwlvc69gsc39c59l18gud60c59n20kqg53.com
id:0002 : h54i25l28i55b28m19l68gvb38o21orh34nwgtnrir.net
id:0003 : fro51owbyhsb48csi65avm39bqf22lygsjzmw.org
id:0004 : d20bxguf32p32ouavixl28d10cxm29ngp32a37gy.info
```

fig. 6 Generating Zeus domain names

Zeus-DGA

In the case of this Zeus botnet, the parameter for the DGA mechanism is calculated from the current date. The list contains over 1000 domains and changes every 7 days. Each name consists of a string with a length of 32 to 48 chars, and one of TLDs: ru, com, biz, info,net or org. It is worth noting that the domain names do not contain the “-” sign. Below is a regular expression for searching ZeuS domains in log files:

```
[a-z0-9]{32,48}\.(ru|com|biz|info|org|net)
```



Zapora systemu Windows

Ogólne | Wyjątki | Zaawansowane

Zapora systemu Windows blokuje przychodzące połączenia sieciowe z wyjątkiem programów i usług zaznaczonych poniżej. Dodanie wyjątków umożliwia lepsze działanie niektórych programów, ale może zwiększyć zagrożenie bezpieczeństwa.

Programy i usługi:

Nazwa

- ☐ Architektura UPnP
- ☑ Diagnostyka sieci dla systemu Windows XP
- ☑ Eksplorator Windows
- ☑ Launch TightVNC Server
- ☑ Opera Internet Browser
- ☑ Pomoc zdalna
- ☑ Pulpit zdalny
- ☑ TCP 26021
- ☐ Udostępnianie plików i drukarek
- ☑ UDP 23048

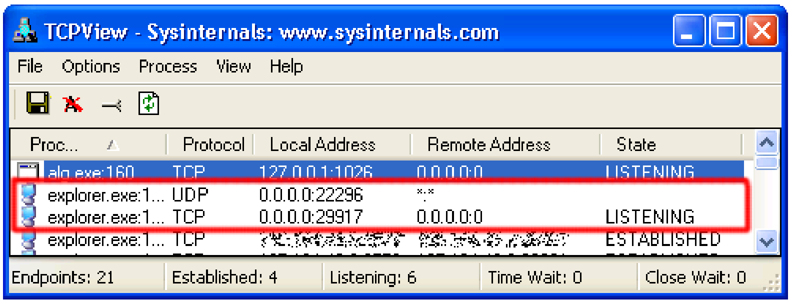
☑ Wyświetl powiadomienie, gdy Zapora systemu Windows zablokuje program

Jakie ryzyko wiąże się z zezwalaniem na wyjątki?

fig.7: Added exceptions in the configuration of the system firewall

How to recognize a new ZeuS infection?

The presence of a new variant of Zeus on the computer can be identified primarily by monitoring network traffic. As shown in Figure 8 – by using TCPView, it is possible to see the new open TCP and UDP ports of theexplorer.exe process. In addition – to allow communication with the ZP2P network – the Trojan adds new rules to the system firewall. As shown in Figure 7, there are the two new exceptions to allow connections to specific TCP and UDP ports. The range of these ports can be read from plot 2 and 3



TCPView - Sysinternals: www.sysinternals.com

```
Proc...          Protocol  Local Address     Remote Address    State
alg.exe:160      TCP       127.0.0.1:1026    0.0.0.0:0         LISTENING
explorer.exe:1...UDP       0.0.0.0:22296     *:*
explorer.exe:1...TCP       0.0.0.0:29917     0.0.0.0:0         LISTENING
explorer.exe:1...TCP       [redacted]        [redacted]        ESTABLISHED
```

Endpoints: 21 | Established: 4 | Listening: 6 | Time Wait: 0 | Close Wait: 0

fig.8: TCP and UDP ports used for P2P communication