

Lots of Virus stuff,

Here is a video of one of the worst public viruses over the last few years, my dad and sister were hit with this one, and our Sys Admin fixed numerous computers because of this exact thing:

<http://www.youtube.com/watch?v=Co2zIsdwXU8>

Script that is basically an irritating virus:

```
@echo off  
  
:s  
  
start cmd.exe  
  
got :s
```

Another, this can really scare people:

```
shutdown -t 1000 -s
```

Ten hackers that have taken real jobs:

<http://www.eweek.com/c/a/Security/10-Notorious-Hackers-Who-Went-to-Work-for-The-Man-304218/>

Zero-day attacks occur during the vulnerability window that exists in the time between when a vulnerability is first exploited and when software developers start to develop and publish a counter to that threat.

For [viruses](#), [Trojans](#) and other zero-day attacks, the vulnerability window follows this time line:

- The developer creates software containing an unknown vulnerability
- The attacker finds the vulnerability before the developer does
- The attacker writes and distributes an exploit while the vulnerability is not known to the developer
- The developer becomes aware of the vulnerability and starts developing a fix.

TOP TEN COSTLY VIRUSES TO DATE (many of these are duplicated on the top ten most famous viruses in the other notes).

1. MyDoom

The most devastating computer virus to date is MyDoom, which caused over \$38 billion in damages. In addition to being the most expensive virus to date, its effects were far-reaching and fast-moving. When a user was infected with the virus it creates network openings which allowed others to have access to your computer. In addition, the virus also had the ability to open random programs. In 2004, an estimated 25% of all emails had been infected by the virus.

2. SoBig

Another harmful and expensive computer virus is SoBig. In 2003, the SoBig virus caused over \$37.1 billion in devastation. This fast-spreading virus circulated through email as viral spam, and if exposed, the virus had the capability to copy files, emailing itself to others and causing serious damage to [computer software](#) and hardware.

3. ILOVEYOU

ILOVEYOU is another particularly malicious virus that spread quickly through email, websites and file sharing. The ILOVEYOU virus, or the "Love Letter" worm, affected more than 500,000 systems in 2000 and produced over \$15 billion in damages, including \$5.5 billion in the first week alone. The virus replicated itself and exposed itself to everyone in the owner's contact list. This virus was a pioneer for other viruses, as it was one of the first to attach to an email.

4. Conficker

The Conficker virus caused over \$9.1 billion in damages in 2007 and infected millions of [computers](#) around the world. The virus scanned computers for weaknesses and vulnerabilities, logged keystrokes and downloaded code from hacker-selected websites and more.

5. Code Red

One of the most well-known viruses to date is the Code Red virus. It caused over \$2 billion in damages in 2001, and [had the ability to break into computer networks](#) and exploit weaknesses in [Microsoft software](#). Once the virus infected the machine, it actively looked for other machines on the networks to attack.

6. Melissa

The Melissa virus was a particularly slimy virus that sent out infected Microsoft Word documents through Microsoft Outlook, delivering viral messages to everyone listed in the Outlook address book. The messages appeared to be coming from the Outlook owner, but was really the Melissa virus at work. A tell-tale indicator that Melissa had infiltrated your Outlook is if your contacts had received an email from you with the message: "Here is that document you asked for ... don't show anyone else." There would be a word document attached, complete with the Melissa virus. In 1999, Melissa caused \$1.2 billion in damages.

7. SirCam

SirCam was a worm that caused over \$1 billion in damages in 2001. This virus had the

ability to compromise confidential information, delete items or use up space on your hard drive until there was not enough memory to store anything else.

8. SQL Slammer

SQL Slammer is a virus that greatly affected banks and caused Internet speed to lag significantly across the globe. SQL Slammer caused an estimated \$750 million in damages in 2003, and affected 200,000 computers worldwide.

9. Nimda

Nimda is one of the Internet's most widespread viruses and among the costliest as well. The virus caused \$635 million worth of damages in 2001 and caused Internet browsing time to slow significantly. Additionally, it could affect a user's email account and send out a read-me file to all contacts listed in the email address book. The virus caused traffic and Internet speeds to slowdown.

10. Sasser

Sasser created quite a bit of trouble in 2004 when it piled up \$500 million in damages, devastated the British Coast Guard mapping system and caused numerous canceled flights. The creator of Sasser was identified as a teenager from Germany, and was quickly apprehended when one of his "friends" turned him in for a \$250,000 bounty posted by Microsoft.

The Bottom Line

While the Internet can be a wonderful resource for doing everything from [communicating](#) with friends and colleagues to checking your bank statement, it is not necessarily the safest of places to perform such transactions when viruses are lurking in the midst. [Protect yourself and your computer](#) with quality anti-virus software, and continue to browse safely on the Internet.

10. The **Sobig Worm** was a [computer worm](#) that infected millions of [Internet](#)-connected, [Microsoft Windows](#) computers in August 2003.

Although there were indications that tests of the worm were carried out as early as August 2002, **Sobig.A** was first found in the wild in January 2003. **Sobig.B** was released on May 2003. It was first called *Palyh*, but was later renamed to Sobig.B after [anti-virus](#) experts discovered it was a new generation of Sobig. **Sobig.C** was released May 31 and fixed the timing bug in Sobig.B. **Sobig.D** came a couple of weeks later followed by **Sobig.E** on June 25. On August 19, **Sobig.F** became known and set a record in sheer volume of e-mails.

The worm was most widespread in its "Sobig.F" variant.

Sobig is not a [computer worm](#) in the sense that it replicates by itself, but also a [Trojan horse](#) in that it masquerades as something other than [malware](#). The Sobig worm will appear as an [electronic mail](#) with one of the following subjects:

CIA (Possible) Techniques (2):

Flame:

Flame,^[a] also known as **Flamer**, **sKyWiper**,^[b] and **Skywiper**,^[2] is modular computer [malware](#) discovered in 2012^{[3][4]} that attacks computers running the [Microsoft Windows](#) operating system.^[5] The program is being used for targeted [cyber espionage](#) in [Middle Eastern](#) countries.^{[1][5][6]}

Its discovery was announced on 28 May 2012 by MAHER Center of Iranian National [Computer Emergency Response Team](#) (CERT),^[5] [Kaspersky Lab](#)^[6] and [CrySyS Lab](#) of the [Budapest University of Technology and Economics](#).^[1] The last of these stated in its report that it "is certainly the most sophisticated malware we encountered during our practice; arguably, it is the most complex malware ever found."^[1]

Flame can spread to other systems over a [local network](#) (LAN) or via [USB stick](#). It can record audio, [screenshots](#), [keyboard activity](#) and [network traffic](#).^[6] The program also records [Skype](#) conversations and can turn infected computers into [Bluetooth](#) beacons which attempt to download contact information from nearby Bluetooth-enabled devices.^[7] This data, along with locally stored documents, is sent on to one of several command and control servers that are scattered around the world. The program then awaits further instructions from these servers.^[6]

According to estimates by Kaspersky in May 2012, Flame had initially infected approximately 1,000 machines,^[7] with victims including governmental organizations, educational institutions and private individuals.^[6] At that time 65% of the infections happened in Iran, Israel, Sudan, Syria, Lebanon, Saudi Arabia, and Egypt,^{[3][6]} with a "huge majority of targets" within Iran.^[8] Flame has also been reported in Europe and North America.^[9] Flame supports a "kill" command which wipes all traces of the malware from the computer. The initial infections of Flame stopped operating after its public exposure, and the "kill" command was sent.^[10]

Flame is an uncharacteristically large [program](#) for malware at 20 [megabytes](#). It is written partly in the [Lua](#) scripting language with compiled [C++](#) code linked in, and allows other attack modules to be loaded after initial infection.^{[6][13]} The malware uses five different encryption methods and an [SQLite](#) database to store structured information.^[1] The method used to inject code into various processes is stealthy, in that the malware modules do not appear in a listing of the modules loaded into a process and malware [memory pages](#) are protected with READ, WRITE and EXECUTE [permissions](#) that make them inaccessible by user-mode applications.^[1] The internal code has few similarities with other malware, but exploits two of the same security vulnerabilities used previously by Stuxnet to infect systems.^{[c][1]} The malware determines what [antivirus software](#) is installed, then customises its own behaviour (for example, by changing the [filename extensions](#) it uses) to reduce the probability of detection by that software.^[1] Additional indicators of compromise include [mutex](#) and [registry](#) activity, such as installation of a fake audio driver which the malware uses to maintain persistence on the compromised system.^[13]

Flame is not designed to deactivate automatically, but supports a "kill" function that makes it eliminate all traces of its files and operation from a system on receipt of a module from its controllers.^[7]

Flame was signed with a fraudulent [certificate](#) purportedly from the Microsoft Enforced Licensing Intermediate PCA certificate authority.^[14] The malware authors identified a Microsoft [Terminal Server](#) Licensing Service certificate that inadvertently was enabled for code signing and that still used the weak [MD5 hashing algorithm](#), then produced a counterfeit copy of the certificate that they used to [sign](#) some components of the malware to make them appear to have originated from Microsoft.^[14] A successful [collision attack](#) against a certificate was previously demonstrated in 2008,^[15] but Flame implemented a new variation of the chosen-prefix collision attack.^[16]

Deployment

Like the previously known cyber weapons [Stuxnet](#) and [Duqu](#), it is employed in a targeted manner and can evade current security software through [rootkit](#) functionality. Once a system is infected, Flame can spread to other systems over a local network or via USB stick. It can record audio, screenshots, keyboard activity and [network traffic](#).^[6] The program also records Skype conversations and can turn infected computers into Bluetooth beacons which attempt to download contact information from nearby Bluetooth enabled devices.^[7] This data, along with locally stored documents, is sent on to one of several command and control servers that are scattered around the world. The program then awaits further instructions from these servers.^[6]

Unlike Stuxnet, which was designed to [sabotage](#) an industrial process, Flame appears to have been written purely for [espionage](#).^[17] It does not appear to target a particular industry, but rather is "a complete attack toolkit designed for general cyber-espionage purposes".^[18]

Using a technique known as [sinkholing](#), Kaspersky demonstrated that "a huge majority of targets" were within Iran, with the attackers particularly seeking [AutoCAD](#) drawings, [PDFs](#), and [text files](#).^[8] Computing experts said that the program appeared to be gathering technical diagrams for intelligence purposes.^[8]

A network of 80 servers across Asia, Europe and North America has been used to access the infected machines remotely.^[19]

Origin

On June 19, 2012, [The Washington Post](#) published an article claiming that Flame was jointly developed by the U.S. [National Security Agency](#), [CIA](#) and Israel's military at least five years prior. The project was said to be part of a classified effort code-named [Olympic Games](#), which was intended to collect intelligence in preparation for a cyber-sabotage campaign aimed at slowing Iranian nuclear efforts.^[20]

According to Kaspersky's chief malware expert, "the geography of the targets and also the complexity of the threat leaves no doubt about it being a nation-state that sponsored the research that went into it."^[3] Kaspersky initially said that the malware bears no resemblance to Stuxnet, although it may have been a parallel project commissioned by the same attackers.^[21] After analysing the code further, Kaspersky later said that there is a strong relationship between Flame and Stuxnet; the early version of Stuxnet contained code to propagate via USB drives that is nearly identical to a Flame module that exploits the same [zero-day vulnerability](#).^[22]

Iran's CERT described the malware's encryption as having "a special pattern which you only see coming from Israel".^[23] [The Daily Telegraph](#) reported that due to Flame's apparent targets—which included Iran, Syria, and the [West Bank](#)—Israel became "many commentators' prime suspect". Other commentators named [China](#) and the U.S. as possible perpetrators.^[21] [Richard Silverstein](#), a commentator critical of Israeli policies, stated that he had confirmed with a "senior Israeli source" that the malware was created by Israeli computer experts.^{[21][24]} [The Jerusalem Post](#) wrote that Israel's Vice Prime Minister [Moshe Ya'alon](#) appeared to have hinted that his government was responsible,^[21] but an Israeli spokesperson later denied that this had been implied.^[25] Unnamed Israeli security officials suggested that the infected machines found in Israel may imply that the virus could be traced to the U.S. or other Western nations.^[26] The U.S. has officially denied responsibility.^[27]

Rootkit:

rootkit installation can be automated, or an [attacker](#) can install it once they've obtained root or Administrator access. Obtaining this access is either a result of direct attack on a system (i.e. exploiting a known vulnerability, password (either by [cracking](#), [privilege escalation](#), or [social engineering](#))). Once installed it becomes possible to hide the intrusion as well as to maintain privileged access. Like any software they can have a good purpose or a malicious purpose. The key is the root/Administrator access. Full control over a system means that existing software can be modified, including software that might otherwise be used to detect or circumvent it.

Rootkits and their payloads have many uses:

- Provide an attacker with full access via a [backdoor](#), permitting unauthorized access to, for example, steal or falsify documents. One of the ways to carry this out is to subvert the login mechanism, such as the /bin/login program on [Unix-like](#) systems or [GINA](#) on Windows. The replacement appears to function normally, but also accepts a secret login combination that allows an attacker direct access to the system with administrative privileges, bypassing standard [authentication](#) and [authorization](#) mechanisms.
- Conceal other [malware](#), notably password-stealing [key loggers](#) and [computer viruses](#).^[17]
- Appropriate the compromised machine as a [zombie computer](#) for attacks on other computers. (The attack originates from the compromised system or network, instead of the attacker's system.) "Zombie" computers are typically members of large [botnets](#) that can launch [denial-of-service attacks](#) and distribute [e-mail spam](#).
- Enforcement of [digital rights management](#) (DRM).

In some instances, rootkits provide beneficial functionality, and may be installed intentionally by the computer owner:

- Conceal [cheating in online games](#) from software like [Warden](#).^[18]
- Detect attacks, for example, in a [honeypot](#).^[19]

- Enhance emulation software and security software.^[20] [Alcohol 120%](#) and [Daemon Tools](#) are commercial examples of non-hostile rootkits used to defeat copy-protection mechanisms such as [SafeDisc](#) and [SecuROM](#). [Kaspersky](#) antivirus software also uses techniques resembling rootkits to protect itself from malicious actions. It loads its own [drivers](#) to intercept system activity, and then prevents other processes from doing harm to itself. Its processes are not hidden, but cannot be terminated by standard methods (It can be terminated with [Process Hacker](#)).
- Anti-theft protection: Laptops may have BIOS-based rootkit software that will periodically report to a central authority, allowing the laptop to be monitored, disabled or wiped of information in the event that it is stolen.^[21]
- Bypassing [Microsoft Product Activation](#)^[22]

Types

Honeypots can be classified based on their deployment and based on their level of involvement. Based on deployment, honeypots may be classified as:

1. production honeypots
2. research honeypots

Production honeypots are easy to use, capture only limited information, and are used primarily by companies or corporations; Production honeypots are placed inside the production network with other production servers by an organization to improve their overall state of security. Normally, production honeypots are low-interaction honeypots, which are easier to deploy. They give less information about the attacks or attackers than research honeypots do.

Research honeypots are run to gather information about the motives and tactics of the [Blackhat](#) community targeting different networks. These honeypots do not add direct value to a specific organization; instead, they are used to research the threats organizations face and to learn how to better protect against those threats.^[1] Research honeypots are complex to deploy and maintain, capture extensive information, and are used primarily by research, military, or government organizations.

Based on design criteria, honeypots can be classified as

1. pure honeypots
2. high-interaction honeypots
3. low-interaction honeypots

Pure honeypots are full-fledged production systems. The activities of the attacker are monitored using a casual tap that has been installed on the honeypot's link to the network. No other software needs to be installed. Even though a pure honeypot is useful, stealthiness of the defense mechanisms can be ensured by a more controlled mechanism.

Blackholing and sinkholing

With blackholing, all the traffic to the attacked DNS or IP address is sent to a "black hole" (null interface, non-existent server, ...). To be more efficient and avoid affecting network connectivity, it can be managed by the ISP.^[23]

Sinkholing routes to a valid IP address which analyzes traffic and rejects bad ones. Sinkholing is not efficient for most severe attacks.

Stuxnet

is a highly sophisticated [computer worm](#). Discovered in June 2010, Stuxnet initially spreads via [Microsoft Windows](#), and targets [Siemens industrial](#) software and equipment. While it is not the first time that hackers have targeted industrial systems,^[1] it is the first discovered [malware](#) that spies on and subverts industrial systems,^[2] and the first to include a [programmable logic controller](#) (PLC) [rootkit](#).^{[3][4]}

The worm initially spreads indiscriminately, but includes a highly specialized malware payload that is designed to target only Siemens [supervisory control and data acquisition](#) (SCADA) systems that are configured to control and monitor specific industrial processes.^{[5][6]} Stuxnet infects PLCs by subverting the [Step-7](#) software application that is used to reprogram these devices.^{[7][8]}

Different variants of Stuxnet targeted five Iranian organizations,^[9] with the probable target widely suspected to be [uranium enrichment](#) infrastructure in [Iran](#).^{[8][10][11]} [Symantec](#) noted in August 2010 that 60% of the infected computers worldwide were in Iran.^[12] Siemens stated on 29 November that the worm has not caused any damage to its customers,^[13] but the [Iran](#) nuclear program, which uses [embargoed](#) Siemens equipment procured secretly, has been damaged by Stuxnet.^{[14][15]} [Kaspersky Lab](#) concluded that the sophisticated attack could only have been conducted "with nation-state support".^[16] This was further supported by the [F-Secure](#)'s chief researcher [Mikko Hyppönen](#) who commented in a Stuxnet FAQ, "That's what it would look like, yes".^[17] It has been speculated that [Israel](#)^[18] and the [United States](#) may have been involved.^{[19][20]}

In May 2011, the PBS program [Need To Know](#) cited a statement by Gary Samore, White House Coordinator for Arms Control and Weapons of Mass Destruction, in which he said, "we're glad they [the Iranians] are having trouble with their centrifuge machine and that we – the US and its allies – are doing everything we can to make sure that we complicate matters for them", offering "winking acknowledgement" of US involvement in Stuxnet.^[21] According to [The Daily Telegraph](#), a showreel that was played at a retirement party for the head of the [Israel Defense Forces](#) (IDF), [Gabi Ashkenazi](#), included references to Stuxnet as one of his operational successes as the IDF chief of staff.^[18]

On 1 June 2012, an article in *The New York Times* said that Stuxnet is part of a U.S. and Israeli intelligence operation called "[Operation Olympic Games](#)", started under President [George W. Bush](#) and expanded under President [Barack Obama](#).^[22]

CountryInfected computers

Iran58.85%

Indonesia18.22%

India8.31%

Azerbaijan2.57%

United States1.56%

Pakistan1.28%

Others9.2%

Operation

“ [O]ne of the great technical blockbusters in malware history. ”

—*Vanity Fair*, April 2011^[24]

Unlike most malware, Stuxnet does little harm to computers and networks that do not meet specific configuration requirements; "The attackers took great care to make sure that only their designated targets were hit...It was a marksman's job."^[31] While the worm is promiscuous, it makes itself inert if Siemens software is not found on infected computers, and contains safeguards to prevent each infected computer from spreading the worm to more than three others, and to erase itself on 24 June 2012.^[24]

For its targets, Stuxnet contains, among other things, code for a [man-in-the-middle attack](#) that fakes industrial process control sensor signals so an infected system does not shut down due to abnormal behavior.^{[24][31][20]} Such complexity is very unusual for [malware](#). The worm consists of a layered attack against three different systems:

1. The [Windows operating system](#),
2. Siemens PCS 7, WinCC and STEP7 industrial software applications that run on Windows and
3. One or more Siemens S7 PLCs.

In April 2011 Iranian government official Gholam Reza Jalali stated that an investigation had concluded that the United States and Israel were behind the Stuxnet attack.^[113] According to *Vanity Fair*, Rieger stated that three European countries' intelligence agencies agreed that Stuxnet was a joint United States-Israel effort. The code for the Windows injector and the PLC payload differ in style, likely implying collaboration. Other experts believe that a US-Israel cooperation is unlikely because "the level of trust between the two countries' intelligence and military establishments is not high."^[24]

[China](#),^[114] [Jordan](#), and [France](#) are other possibilities, and Siemens may have also participated.^{[24][107]} Langner speculated that the infection may have spread from USB drives belonging to Russian contractors since the Iranian targets were not accessible via the internet.^{[8][115]}

Sandro Gaycken from the Free University Berlin argued that the attack on Iran was a ruse to distract from Stuxnet's real purpose. According to him, its broad dissemination in more than 100,000 industrial plants worldwide suggests a field test of a cyber weapon in different security cultures, testing their preparedness, resilience, and reactions, all highly valuable information for a cyberwar unit.^[116]

The [United Kingdom](#) has denied involvement in the virus's creation.^[117]

Stratfor Documents released by Wikileaks suggest that the International Security Firm 'Stratfor' believe that Israel is behind Stuxnet - "But we can't assume that because they did stuxnet that they are capable of doing this blast as well."^[118]

Another virus of abnormal beginnings.....

Duqu

On 1 September 2011, a new worm was found, thought to be related to Stuxnet. The Laboratory of Cryptography and System Security (CrySyS) of the [Budapest University of Technology and Economics](#) analyzed the malware, naming the threat **Duqu**.^{[119][120]} [Symantec](#), based on this report, continued the analysis of the threat, calling it "nearly identical to Stuxnet, but with a completely different purpose", and published a detailed technical paper.^[121] The main component used in Duqu is designed to capture information^[20] such as keystrokes and system information. The exfiltrated data may be used to enable a future Stuxnet-like attack. On 28 December 2011, [Kaspersky Lab](#)'s director of global research and analysis spoke to [Reuters](#) about recent research results showing the platform, Stuxnet and Duqu both were built on, originated in 2007 and is being referred to as Tilded, due to the ~d at the beginning of the file names. Also uncovered in this research was the possibility for three more variants based on the Tilded platform.^[122]

The term Duqu is used in a variety of ways:

- **Duqu malware** is a variety of software components that together provide services to the attackers. Currently this includes information stealing capabilities and in the background, kernel drivers and injection tools. Part of this malware is written in unknown high level programming language,^[5] dubbed "Duqu framework". It is not C++, Python, Ada, Lua and many other checked languages. However, **recent evidence** suggests that Duqu may have been written in Object Oriented C (OO C) and compiled in Microsoft Visual Studio 2008.
- **Duqu flaw** is the flaw in Microsoft Windows that is used in malicious files to execute malware components of Duqu. Currently one flaw is known, a TTF related problem in win32k.sys.
- **Operation Duqu** is the process of only using Duqu for unknown goals. The operation might be related to Operation Stuxnet.