Security Measures Against Malware Penetration and Spreading

[1]Markina T.A., [1]Ogolyuk A.A.

[1]*St. Petersburg National research university of Information technologies, mechanics and optics, Kronverkskiy pr., 49, Saint Petersburg, Russia*

Abstract – As the title implies the article describes the problems of protection against malicious programs. The widely used method of protection, which is currently the signature analysis, is described. Also the article considers the causes of low efficiency of this method/ A method of protection against malicious programs that would guarantee to prevent the recording and execution of unauthorized programs is offered.

Keywords – information security, computer protection, antivirus, malware, security method.

1. Malware – the modern main threat

Today the one of the most dangerous threats is computer viruses and penetration attacks using malware. They mostly serve to allow unauthorized access to the information stored on computers. Multiply data on world known companies annual financial losses prove this fact.

For example G Data Software researches declare annual malware threats number increasing. Total number of threats makes the annual record. The total number of threats did increase on 50% comparing to previous year. In other words the number of malware appeared in the first half of this year is already more than total malware appeared in the full previous year. G Data Software forecast for next year is more than 2 000 000 of malware applications.

Previous statistics are 39 670 in 2007, 133 253 in 2008, 894 250 in 2009, 1 588 053 in 2010. [1] We can see that the number of threads grows exponentially. “Malware industry produces about 4 new viruses per minute” – says Ralf Bansmuller (G Data Security Labs CEO)

If we transform the statistics of malware threads we can calculate that Anti Virus companies need to analyze about 10 000 new malware applications per day. Sure such amount can’t be managed only using human based analysis. To analyze this enormous amount of new malware the new forms of automatic analysis is needed. But unfortunately such automatic tools are not yet effective enough to fully research modern malware examples and without human help are useless enough. Making extrapolation (basing on current statistics) we can assume that in 2014 there will be about 50 000 000 malware application available. In 2015 it can be more than 200 000 000 and so on. Even if someone doesn’t see the danger in this number, we can calculate the number of malware that undetected pass through antivirus control. If we assume Anti Virus Software effectiveness as 99% (that is very optimistic forecast) then 1% of undetected malware was 16 000 in 2008 and in 2014 it will be already more than half million.

While real researches declare detection of just 45-55% of newly created malware applications [2]. I.e. the real effectiveness of Anti Virus Software while detecting new malware is much less than above assumption. That brings us to enormous number of undetected malware applications in near years.

2. Common antimalware methods

Most modern antimalware methods base on malware detection or malware penetration preventing. The most used methods are:

- *Malware scanning* (signature based). The main principle is to scan all files, boot sectors and memory space to detect malware presence (find unique malware executable code)

- *Static analysis*. Similar to previous one but uses executable code decompilation and comparing to signatures only while process execution.

- *Dynamic analysis*. Differs from static one by decompilation absence and analyses process behavior while executing.

- *Anti Visrus monitor*. Similar to scanner, but is always present in the memory and works in the background scanning files in real time while this objects are accessed (files, boot sectors, etc.). Starts on OS loading and automatically checks every object. Most monitors use File System driver to intercept (and scan) file objects access.

- *Finding differences*. This method is based on calculating "fingerprints" (CRC-sum) of files and system sectors. This "fingerprints" are stored in local database. On next run the scanner compares current "fingerprints" with original ones and alerts the user.

- *Immunizer*. This method has two implementations. First one writes additional executable code to the file ending. And while file is executed checks its integrity. Second one protects against certain virus modifying the file in such way that virus thinks this file is already infected.

- *Behavior analyzer*. In this case analyzer code is always present in computer memory and intercepts any system events. When it finds "suspicious" events similar to malware it blocks such actions or asks user to confirm it.

Modern Anti Virus software mostly use all methods described above.

3. Common antimalware methods limitations

Let's find the source of such critical situation with malware while all described methods are implemented.

Signature method limitations are big number of signatures and the need of updates. Such big signature databases slow down the scan process and are difficult in maintaining and updating.

Scanners most important limitation is an inability to detect malware modifications. For example there are more than 10 modifications of Melissa malware application and every new modification was followed by signature updates. And this signature updates can take a long time.

This brings us to second limitation: while new signature update is unavailable the user is completely unprotected against new malware modification.

And the third scanner limitation is working only by request (i.e. when user launches the scanner). User can forget to run scanner while downloading files from Internet or from removable drive.

Anti Virus software developers declare that heuristic analysis method embedded into Anti Virus software can help, but it gives too many false positives. Independent tests of such methods show that heuristics can detect only 40-50% of malware applications.

Also we must not forget that scanning process slows down the system. It takes memory and CPU time, this can be very important on old computers where the speed can drop up to 100-300%.

Difference detectors also have limitations. They can't find malware while penetrating, detecting malware code only after spreading and system modifications. Also they can't detect malware in newly created files (in emails, removable media, etc.) due to CRC absence for new objects.

Immunizers have only one but big limitation. They can protect only against certain malware not all at once. That is why immunizer method is less present in modern world.

Different methods of encryption and packing of malware make them invisible to modern Anti Virus software. To detect such malware there is a need for strong unpacking and decrypting technologies that are rarely used.

So for today the most used methods to protect against malware penetration have low enough effectiveness.

4. Implementing new method of antimalware protection

Reviewing described above limitations we suggest another method to protect against malware. This method is based on objects access control policy. Method declares prevention of malware penetration and unauthorized applications installation or execution. Also it declares deny of execution from those resources that are uncontrolled during installation phase (such as Internet, removable drives, etc.).

This method uses special access "subject". In common case we can use "all processes" subject (both system and user ones). All access control policies are implemented to this "subject". I.e. we can create profile named "All processes" which will contain all file objects access control limitations. (Sure in more complex situation there can be different profiles for different types of system and user processes).

As an access objects this method uses only files with predefined extensions (that cover all executable files, for example *.exe extension).

To remove the limit of what malware application is penetrating the computer (newly created or already existed) we define File System "Objects" (FSO) that can be executed.

The short extension list can be like this:

- *.com,
- *.exe,
- *.bat,
- *.cmd,
- *.php,
- *.vbs,
- *.vbe,
- *.js,
- *.jse,
- *.wsf,
- *.wsh,
- etc. [5].

Then we define file system objects which are needed for Operation System and user applications correct working.

Mostly "objects" are stored in such system folders:

- %Program Files%,
- %windir% or
- %systemroot%

and have such extensions:

- *.exe,
- *.bat,
- *.cmd,
- *.dll,
- *.drv,
- *.ocx,
- *.config,
- *.manifest,
- *.log,

etc.

To prevent unauthorized applications executing we allow executing only those objects that have "executable" extensions. Also we grant read access to these objects.

To prevent malware penetration we deny the modification of all defined objects (executables) and deny to create new one executable objects (including rename operation). These basic policies above can be changed to allow correct application working and new authorized applications installation.

All above rules are set to all system users including Administrators users and Administrator group.

We did implement this method in test software (for Windows XP, Windows 2003, Windows 7 x86, Windows 7 x64 and Windows 2008 R2; all test were performed using VMware® Workstation 6 virtual machines) to research its effectiveness against different types of malware applications.

As there are too many kinds of malware applications, we choose for initial test the only one type – "drive-by" download type of malware applications.

Mostly this type of malware is downloaded from social web sites using script exploits or redirection code (cross site references). In common scenario the malware is downloaded to target computer automatically without user confirmation. This is possible due exploiting browser, plug-ins and ActiveX elements faults. Common scenario uses entering infected site and then redirects (with IFrame usage) to the malicious page containing CSS and malware script loader. The number of intermediate site redirections can be more than one.

We did test the 4 redirectors from TOP 20 (2011) entries of Kaspersky Lab Anti Virus Company.
They are:

- Trojan-Downloader.JS.Agent.fzn (12),
- Trojan-Downloader.JS.Agent.gay (13),
- Trojan-Downloader.JS.Iframe.cfw (14),
- Trojan-Downloader.JS.IFrame.tm (15).

They use tag <iframe> and redirect the user on the malicious web-site and download another JavaScript file, which starts exploits. Script-downloaders are of two groups:

- Trojan.JS.Redirector.pz (5),
- Trojan.JS.Redirector.qa (7),
- Trojan.JS.Redirector.py (8),
- Trojan.JS.Redirector.qb (9),
- Trojan-Downloader.JS.Agent.gbj (11),
- Trojan-Downloader.JS.Agent.gaf (19). [7].

All this malware downloader's use html tags. They use <img> tag fields and tag fields. Possibly this methods is implemented to bypass Anti Virus control emulators and "sandboxes" which doesn't properly support JavaScript and HTML integration.

We did use previously described policies to deny installing auto downloaded malware samples and to deny the modification of system files.

As a result of this test all 4 samples of malware downloader's fail to run downloaded code on victim's machine (while without our test protecting software that implements described above method all 4 malware sample successfully accomplished to infect target machines). This result shows good effectiveness of suggested method.

The benefits of this method are:

- simplicity;
- protecting against both existing and newly created malware;
- null false positives;
- no need of signatures database;
- low resource requirements (memory and CPU).

Also we plan to research it effectiveness on other types of malware samples together with researches of performance parameters (to prevent protected system performance degradation common while using traditional Anti Virus solutions)

References

[1]. The number of new viruses in 2010 is close to 2 000000, http://www.cybersesecurity.ru/crypto/103172.html

[2]. The results of test of proactive antivirus protection (June 2010), http://www.antimalware.ru/proactive_test_2010

[3]. Kaspersky E., Zenkin D., Computer viruses: origin, the real threat and protection methods, http://www.nkj.ru/archive/articles/7889/

[4]. Shabanov I. The results of test of antivirus products for performance (February 2010), http://www.antimalware.ru/antivirus_test_performance_2010#part4

[5]. Executable files: extensions, formats, http://open-file.ru/types/executable/

[6]. Sheglov A. U. Protect computer information from unauthorized access - St. Petersburg: Science and Technology, 2004.

[7]. Zakorzhevsky B. Top Viruses - June 2011, http://www.securelist.com/ru/analysis/208020706/rss/analysis/

[8]. Sophos Security Threat Report 2011, http://www.techrepublic.com/whitepapers/sophos-security-threat-report-2011/2405057

Corresponding author: **Markina Tatiana**
Institution: **St. Petersburg National research university of Information technologies, mechanics and optics, Saint Petersburg, Russia**
E-mail: **tanyashibaeva@mail.ru**