FROM TRADITIONAL ANTIVIRUS TO COLLECTIVE INTELLIGENCE

PANDA'S TECHNOLOGY EVOLUTION

Technology Paper by **Panda Research**
research.pandasecurity.com

PANDA SECURITY | *One step ahead.*

Content

PANDA | *One step ahead.*

1. Abstract

There is more malware than ever being released in the wild, and antivirus companies relying on signatures to protect users cannot keep up with the pace of creating signatures fast enough. As a result, the current installed base of anti-malware solutions is proving to be much less effective against the vast amounts of threats in circulation.

As we have been able to prove in a recent research study , even users protected with anti-malware and security solutions with the latest signature database are infected by active malware. Complementary approaches and technologies must be developed and implemented in order to raise the effectiveness to adequate levels.

This paper presents the fourth generation of security technologies by Panda Security, called Collective Intelligence. The Collective Intelligence allows us to maximize our malware detection capacity while at the same time minimizing the resource and bandwidth consumption of protected systems.

The Collective Intelligence represents an approach to security radically different to the current models. This approach is based on an exhaustive remote, centralized and real-time knowledge about malware and non-malicious applications maintained through the automatic processing of all elements scanned.

One of the benefits of this approach is the automation of the entire malware detection and protection cycle (collection, analysis, classification and remediation). However automation in and by itself is not enough to tackle the malware cat-and-mouse game. With large volumes of malware also comes targeted attacks and response time in these scenarios cannot be handled by automation of signature files alone.

The other main benefit that the Collective Intelligence provides is that it allows us to gain visibility and knowledge into the processes running on all the computers scanned by it. This visibility of the community, in addition to automation, is what allows us to tackle not only the large volumes of new malware but also targeted attacks.

Written and reviewed by:
Pedro Bustamante, Senior Research Advisor
Iñaki Urzay, Chief Technology Officer
Luis Corrons, Technical Director PandaLabs
Josu Franco, Director of Corporate Development

2. The malware landscape

I t is a known fact by all security professionals that there are more malware samples infecting users than ever before.

Malware writers have realized they can gain large amounts of money from distributing malware. The shift in motivation for creating malware, combined with the use of advanced techniques, has resulted in an exponential growth of criminally professional malware being created and distributed to infect unsuspecting users.

Also known as a type of targeted attacks, this new malware dynamic has become the next big plague for users and companies alike. Gartner estimates that by the end of 2007 75% of enterprises will be infected with undetected, financially motivated, targeted malware that evaded their traditional perimeter and host defenses[2].

2.1 Antivirus laboratories under attack

Nowadays antivirus laboratories are under a constant and increasingly frequent distributed denial of service attack. The security industry is literally being saturated with thousands of new malware samples every day.

Each one of these new samples needs to be looked at by an analyst trained in reverse engineering in order to create a signature, which is costly and resource intensive from a corporate and business perspective.

Some companies are trying to deal with the problem by increasing the number of

analysts at the labs"[3] or by advocating for stronger intervention[4] by Law Enforcement[5] to help ease the workload by convicting the most active malware creators.
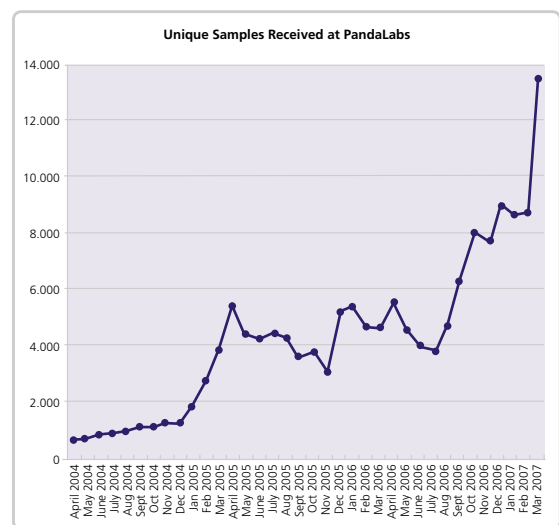
Figure 1:
Unique samples received at PandaLabs 2004 to 2007

Initiatives to get law enforcement more involved are definitely a necessary step in the right direction but unfortunately it seem as an insufficient solution for the short term as the number of variants is increasing incrementally and most of the time only the "mules" and "script kiddies" are actually convicted.

The more advanced malware writers, who are selling their code to spammers, mafias and criminals, are more evasive and harder to catch. In addition, the lack of resources at most law enforcement agencies around the world, tied to insufficient international cooperation and coordination among them make for a difficult task when trying to arrest a suspect or known cyber criminal. In

the long run both a technological and a social approach are needed if we want to solve this problem.

In addition, malware writers are getting sophisticated and reverse engineering some of the latest common threats requires a higher level of knowledge and a larger amount of time dedicated to each sample than historically. Because of this situation antivirus engineers can no longer be employed "by the numbers" to create hundreds of thousands of signatures every few months.

2.2 Malware techniques and design

The main differences between past viruses and today's malware is that the lifecycle has been significantly shortened and the objectives refined; to steal identities, use computers as spam bots, steal online banking credentials, credit card information, web logins, etc.

More importantly, today's malware is designed to not raise any alarms. Unlike in the past where viruses and worms were designed to spread to as many computers as possible without user intervention, generating a lot of noise and media awareness, today's criminal malware wants to be as inconspicuous as possible. In order to achieve its objective, malware today uses advanced techniques to evade detection and "fly low".

2.2.1 Targeted Attacks: staying below the radar

One of the main strategies used by Targeted Attacks for staying below the radar is to distribute few copies of many variants[6].

In the past a single virus or worm was responsible for infecting hundreds of thousands and even millions of computers. Visibility of these situations was very obvious for antivirus labs.

Nowadays malware only infects a few hundred PCs before updating itself with a new, undetectable variant to avoid detection by regular antivirus signatures. The underlying issue is how does an antivirus lab become aware of such an infection if it is only affecting a handful of users?

2.2.2 Malware QA

An older technique used incrementally by malware today is basic QA testing. This is done by testing each variant against the most common antivirus engines to make sure it goes undetected by the majority of them.

This task is greatly simplified by online-scanning services such as Jotti, VirusTotal, the antivirus vendors' online scanning services[7] and online sandboxing services such as Cwsandbox, Norman and Anubis.

Malware creators also count on customized tools to automate testing of new malware against signatures, heuristics and even behavioral analysis technologies. With these tools malware writers can test the quality of their creations off-line, without risking having the sample sent to the antivirus laboratories via the above-mentioned online scanning services.

The objective of malware QA testing is not so much to avoid detection by all scanners and all proactive techniques (generic signatures, heuristics, behavior analysis, behavior blocking, etc.) but to avoid the majority of them. Given its objective of staying below

the radar it is not worth creating the most undetected malware if it is only going to live for a few hours or days.

2.2.3 Rootkits and sandbox detection techniques

Another common detection evading technique which is gaining momentum[8] is the use of rootkit techniques within Trojan and Spyware samples. When used by malware, rootkits create yet another barrier for being detected, especially as advanced rootkit detection technologies have not yet been deployed to all mass-production security solutions.

It also means that the antivirus laboratories need to spend more time analyzing kernel mode drivers than user-mode samples. For example LinkOptimizer, which has been seen in-the-wild in recent months, is able to determine if the machine it is about to infect has security, debugging or system monitoring tools installed. It also checks if it is running in a Virtual Machine environment. If these checks are matched it silently exits and does nothing.

Labs that depend on VM will have to go through great lengths to be able to install certain LinkOptimizer samples in order to analyze them in depth.

At the time of writing few anti-malware and security suites include some basic form of rootkit detection such as low-level access cross-view against API-level calls, but most have not yet incorporated the more advanced rootkit detection and deactivation techniques found in free, stand-alone anti-rootkit utilities[9].

Overall the use of rootkits by malware creators keeps steadily growing and this has become a problem for antivirus laboratories that approach malware reverse engineering in a traditional manner and need to analyze each sample one by one.

Not only the antivirus labs are having problems with rootkits, but also companies are starting to experience the negative effects of rootkits in business, especially when it is used for corporate espionage[10].

In order to get a better idea of the problem at real user's machines we have gathered all known and unknown rootkit detections by our free utility Panda Anti-Rootkit[11] between the months of December 2006 and June 2007 and mapped the distribution of rootkits within malware in the wild. The resulting "Top 5 rootkits in the wild" are shown in figure 2 below, which shows a great increase in the use of kernel-mode rootkits.

2.2.4 Runtime-packers

Perhaps the most common technique to try to evade detection by anti-malware products is the use of obscure runtime packers with anti-debugging and anti-virtualization techniques.

These types of tools can modify and compress an executable file by encrypting and changing its form from its original format. The final result is a modified executable which, when executed, does exactly the same thing as the original code, but from the outside has a completely different form and therefore evades signature-based detection unless either the engine has the specific unpacking algorithm or it is able to unpack it generically.

Malware writers caught up to this approach and we are now even seeing

Top Rootkit	User-mode	Kernel-mode	Prevalence
1 Beagle.Fu		x	6.20%
2 NaviPromo	x		5.73%
3 Rustock.A		x	1.20%
4 Flush.K	x		1.01%
5 Oddysee.B		x	0.20%

Figure 2: Top Rootkits in the Wild as detected by Panda Anti-Rootkit from December 2006 to June 2007

malware which use either modified versions of known packers or even create their own runtime packing routine specifically for their malware samples[12].

In order to address this problem, Panda's engineers have created both generic packer detectors and generic unpacking algorithms which can detect unknown packers and try to unpack them.

However, a more effective solution will be to at least flag the newly created runtime packers as suspicious altogether. Some off-the-shelf perimeter solutions already do this by default. Even some host-based security solutions are using this approach by flagging these types of samples as malicious as is becoming obvious from the different detection names used by the different anti-malware engines[13].

The impact of such an approach to proactive packer detection is not without cost. While speaking to other anti-malware vendors during the 2007 International Antivirus Testing Workshop in Iceland it became apparent that doing so in corporate environments was a good approach, but vendors with high install base on the consumer market could face such a high wave of false positives that the solution could potentially be worse than the problem itself.

2.2.5 Botnets

According to some studies approximately 11% of computers worldwide are infected by bots, which are responsible for sending up to 80% of all spam[14]. A large portion of money made by cyber criminals stems from botnets.

The control of these large networks of compromised machines is sold or rented to perform certain types of cyber-criminal activities, from sending spam runs, distributed denial of service attacks, renting of proxies, keylogging, pay per click installs, adware installations, stored passwords, man in the middle attacks, etc.

PandaLabs has witnessed on-line wars between different bot gangs to win over hijacked PCs. Even though some evidence suggests that there are many PCs belonging to Fortune 500 companies which are controlled remotely by bot herders to send out spam[15], the reality is that virtually every corner of the Internet is infested by bots.

Even though traditional botnets are controlled via IRC, new P2P and HTTP based botnets which use stronger communication encryption are becoming popular among cyber-criminals in order to evade detection and shutdown.

2.2.6 Staged infection vectors

It's nothing new that most of today's malware has a tendency of using a two-staged attack as its main infection technique, either by exploiting known or zero-day vulnerabilities or by using small downloaders which change very rapidly to avoid detection.

While in the past it would take malware authors weeks or even months to take advantage of a vulnerability as its main infection vector, nowadays its normal to see exploits in the wild for vulnerabilities a couple of days after it is known. Even further, organizations that manage darknets such as Team Cymru are seeing new zero-day exploits in the wild using stealthier techniques for days and weeks before it is widely known and before they are massively used by botnets.

Examples such as GDI, animated cursor and VML vulnerabilities are being exploited by automated infection frameworks such as Web-Attacker[16] and MPack[17], which make use of multiple vulnerabilities to exploit unsuspecting and un-patched users in order to infect them with a Trojan.

Downloaders have also become common practice for two-staged infection techniques. First a small file is executed either via a browser drive-by download or similar exploit. This file is coded with a single objective in mind; download a second file from a URL and execute it. This second file in turn is the true Trojan which ends up infecting the system.

These downloaders have become very advanced. SecuriTeam recently ran a Code Cruncher competition to create the smallest downloader in the world[18]. More recently we are seeing a myriad of graphical tools emerge that simplify the creation of new downloaders[19], even with custom packing techniques to evade detection.

2.2.7 "Malware 2.0"

A current trend in malware creation is that the actual binary that infects the user's PC is "dumb" and the intelligence is "in-the-cloud". The code that resides on the PC has some simple functions that it passes on to a remotely compromised server. The server then returns instructions on what to do. Borrowing the (perhaps overused) "2.0" term from current web trends, we will refer to "Malware 2.0" as _malware which separates its intelligence from its code base_.

PandaLabs has reported the "2.0" approach in banking targeted attack Trojans in order to remotely monitor users' browsing habits and, based on the online banking landing page and authentication scheme, inject some type of HTML code or other. Known banking Trojans such as Limbo/NetHell and Sinowal/Torpig use these techniques quite extensively[20]

Other "2.0" techniques recently used by malware are "server-side-compilation", where the webserver re-compiles a new binary every few hours. Lastly, botnets are using fast-flux DNS networks for improved resistance against take-down efforts. These last techniques are more visible in the recent Storm/Nuwar attacks.

One step ahead.

3. Panda's Technology Evolution

Dealing with this malware situation using a traditional signature approach has not been valid for some years now. A complete Host Intrusion Prevention System (HIPS) with advanced heuristics, deep packet inspection firewall, behavior blocking, behavior analysis and system and application hardening are an absolute must for any security solution, even though the sad reality is that about half the solutions on the market do not have these types of technologies yet.[21]

At Panda we research and develop 100% of our core anti-malware technologies. This dedication to innovation has allowed us to lead the way in proactive technology deployment to the market.

Following a defense-in-depth philosophy, which could be summarized as integrating different protection technology layers at different infrastructure layers, Panda Research, a team dedicated to developing new security technologies, developed a new focus to security protection which is based on the concept of **Collective Intelligence**.

The Collective Intelligence concept is designed to complement Panda's integrated desktop, server and gateway protection to take the battle against today's malware dynamic head on and provide the final complement of Panda's ideal protection model.

Before we dive into explaining Collective Intelligence, let's do a walk-through of the different technology generations on top of which Collective Intelligence is built.

3.1 First Generation: Antivirus

The first generation of antivirus products was purely based on signature detection. This generation of technology occupied most of the 1990's and included polymorphic engines as well as basic rule-based MS-DOS, Win32, Macro and, later on, script heuristics. This period was also marked by the appearance of the first massively used win32 Trojans, such as NetBus and BackOrifice.

3.2 Second Generation: Anti-malware

Starting in 2000 new types of malware started to emerge, with file-less network worms and spyware taking the spotlight causing massive and highly visible epidemics.

Basic antivirus engines evolved to integrate personal firewalls to be able to identify and stop network worms based on packet signatures as well as system cleaners to restore modified Operating System settings such as registry entries, HOST files, Browser Helper Objects, etc. It is within this second generation of technologies that Panda Security integrated the *Smart-Clean* functionality into the anti-malware engine, designed to disinfect and restore the Operating System from a spyware or Trojan backdoor infection.

3.3 Third Generation: Proactive technologies

Panda released TruPrevent® behavioral technologies in 2004 after more than three years of intensive research and development.

Since then, TruPrevent® has evolved into a set of behavioral technologies that are substantially more effective at blocking zero-day malware proactively without any dependency on viral signatures than any other previous effort in such direction. TruPrevent® is constantly adapted to new malware techniques and exploits.

TruPrevent® was designed as an additional protection layer to the anti-malware engine. Currently there are more than 5 million computers running TruPrevent®. All these computers also act as high-interaction honeypot nodes which report to PandaLabs any new malware sample that TruPrevent® flags as suspicious and which is not detected by regular antivirus signatures.

TruPrevent's® approach consists of scanning each item or potential threat using different techniques, carrying out in-depth complementary inspections at the different layers of the infrastructure. The approach to TruPrevent® implementations is modular and therefore can be applied both to desktops and servers to become full-blow integrated Host Intrusion Prevention Systems (HIPS).

As an approximate detail of its effectiveness, about two thirds of the new malware samples received at PandaLabs from our users are now coming from automated submissions from TruPrevent®. Technically TruPrevent® consists of 2 main technologies: **behavioral analysis** and **be-**havioral blocking**, also known as system and application hardening. Before going into each of these let's take a look at the underlying uncloaking layer which makes malware visible to these behavioral technologies.

3.3.1 Uncloaking techniques

As malware has evolved so have the techniques used to evade detection and hide from prying eyes.

To combat these hiding techniques there is an underlying layer of uncloaking technologies common to all of Panda products.

The following techniques are able to inspect any item as deeply as necessary, even if the item is making use of stealth techniques to remain hidden in the system, and pass on the results to the scanning and monitoring technologies:

- *Deep Code Inspection*
- *Generic Unpacking*
- *Native File Access*
- *Rootkit Heuristic*

3.3.2 TruPrevent Behavior Analysis

Codenamed Proteus, it acts as a true last line of defense against new malware executing in the machine that manages to bypass signatures, heuristics and behavior blocking. Proteus intercepts, during runtime, the operations and API calls made by each program and correlates them before allowing the process to run completely. The real-time correlation results in processes being allowed or denied execution based on their behavior alone.

One step ahead.

As soon as a process is executed its operations and API calls are monitored silently by Proteus, gathering information and intelligence about that process's behavior. Proteus exhaustively analyses the behavior and is designed to block the malware as soon as it starts performing malicious actions.

If it is determined as suspicious, the process is blocked and killed before it can carry out all of its actions and prevented from running again.

Unlike other behavioral technologies, Proteus is autonomous and does not present technical questions to the end user ("Do you want to allow process xyz to inject a thread into explorer.exe or memory address abc?"). If Proteus thinks that a program is malicious it will block it without requiring user intervention.

Most users cannot make informed decisions when it comes to security. Some behavioral products throw non-deterministic opinions –or behavioral indecisions- whose effectiveness depends on the user clicking on the right choice. A key functionality of any behavioral technology must be making decisions without user intervention. Anything less is a potential point of failure.

Our internal statistics show that this technology alone is capable of detecting over 80 percent of the malware in the wild without signatures and without generating false positives.

This technology does not require signature updates, as it is based solely on the behavior of applications. A bot would not be a bot if it didn't behave as such, but if it does so it will be detected by this technology, regardless of its shape or name. Several third-party tests have been performed on TruPrevent®.Performing tests for behavioral technologies such as TruPrevent, using real-life malware samples, is time-consuming and it requires a fair amount of expertise in the field. It is without doubt much more challenging than performing on-demand tests of antivirus scanners against a collection of viruses.

The first test was commissioned by Panda and it was performed by ICSALabs, a Division of CyberTrust Corporation, in the fall of 2004. ICSALabs tested the technologies against a set of approximately 100 real malware samples. This first test was designed to verify that the technologies worked against a variety of malware types, rather than to reach a conclusion about the overall effectiveness of the technologies over time.

At the same time, ICSALabs tested TruPrevent® against several sets of legitimate applications, from games to Peer-to-peer packages, but was not able to produce any instance of false positives, despite their efforts in this regard.

Another "early" review by PC Magazine USA concluded that *"TruPrevent blocked two-thirds of a sample of recent worms, viruses, and Trojans based strictly on behavior. Blocked no legitimate programs. No noticeable impact on system performance."*
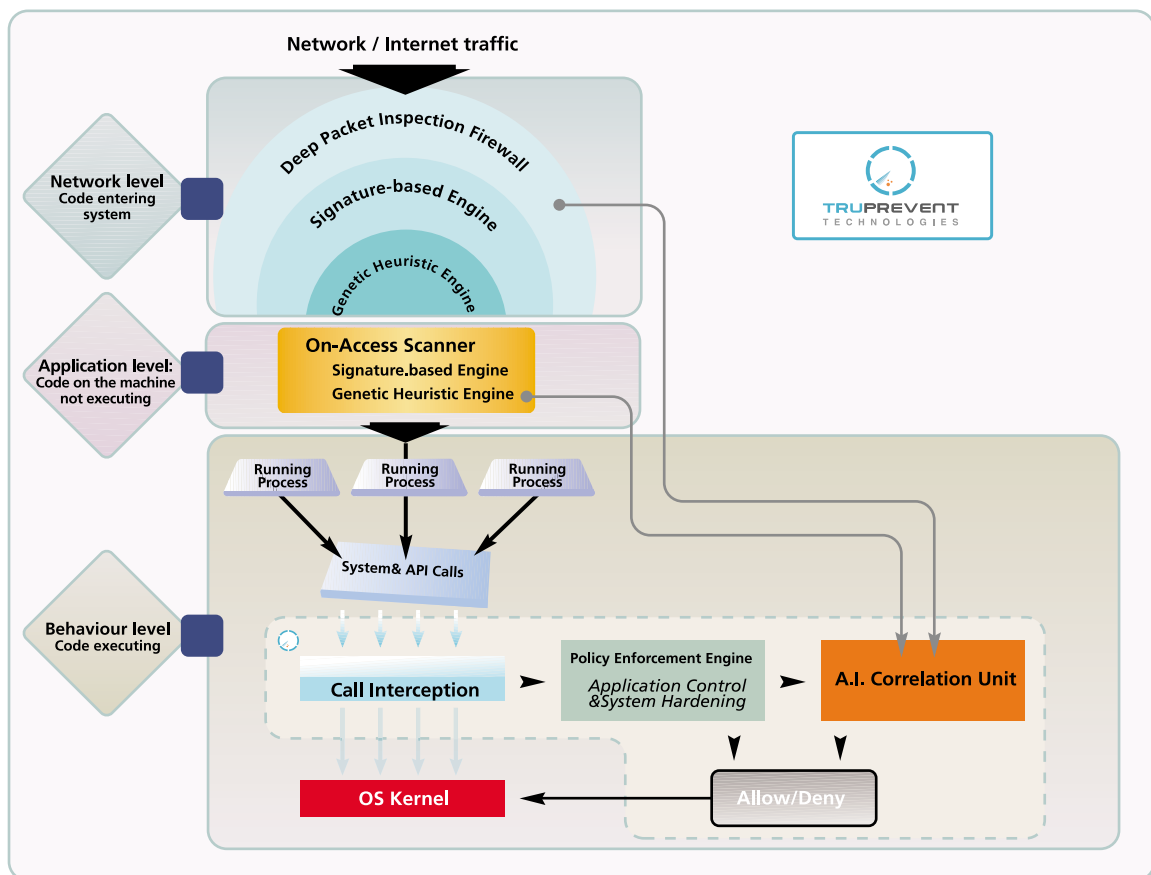
Figure 3: Panda's integrated endpoint security

3.3.3 TruPrevent® Behavior Blocking

Codenamed KRE (Kernel Rules Engine), this is TruPrevent's second main component, also known as Application Control & System Hardening or Resource Shielding.

Hackers and malware abuse the privileges of legitimate applications to attack systems by injecting code. To prevent these types of attacks generically it is very cost-effective to use rule-based blocking technology which can restrict the actions that authorized applications can perform in the system.

KRE is composed of a set of policies which are defined by a set of rules describing allo-wed and denied actions for a particular application of group thereof. Rules can be set to control an application's access to fi-les, user accounts, registry, COM objects, Windows services and network resources.

Despite offering a high degree of granularity to administrators for creating custom policies, the Application Control & System Hardening module (KRE) is shipped with a set of default configuration policies which are managed and updated by PandaLabs.

The default policies provide protection against attacks exploiting common weaknesses found in out-of-the-box as well as fully-patched installations of Windows operating systems.

A recent example of the effectiveness that proactive blocking provides is the never-ending wave of Microsoft Office format vulnerabilities which are being exploited to hide malicious code[24]. These vulnerabilities have been used recently by targeted attacks on certain companies. According to a study of known (patched) and zero-day (un-patched) Microsoft Office vulnerability exploits, an average antivirus signature detection rate of 50% was achieve by all tested antivirus engines. That's a one-in-two chance of being infected by simply opening an exploited Microsoft Word, PowerPoint or Excel document.

On the contrary, behavioral blocking technologies such as TruPrevent, proactively prevents Microsoft Word, PowerPoint, Excel, Access, Acrobat Reader, Windows Media Player and other applications from dropping and running any type of executable code on the system. Unlike any antivirus signatures tested, TruPrevent® provides real zero-day protection against any Microsoft Office exploit, known or unknown.

3.3.4 Genetic Heuristics

"Genetic" technologies are inspired by the field of genetics in biology and its useful-ness to understand how organisms are in-dividually identified and associated to other organisms. These technologies are based on the processing and interpretation of "digital genes", which are represented in our case by quite a few hundred characte-ristics of each file that is scanned.

Codenamed Nereus, the Genetic Heuristic Engine was initially released in 2005. The objective of GHE is to correlate the genetic traits of files by using a proprietary algorithm. The genetic traits define the potential of the software to carry out malicious or harmless actions when executed on a computer. GHE is capable of determining whether a file is innocuous, worm, spyware, Trojan, virus, etc. by correlating the different traits of each item scanned.

GHE can be set to low, medium or high sensitivity with the obvious combination trade-off between detection rates and false positives. The different sensitivity levels are designed to be applied to different environments depending on the probability of malware prevalence on each.

For example at network SMTP gateways we have found that the likelihood of an executable files being malware is very high. Therefore the implementation we have done in our commercial products is of high sensitivity for network layer e-mail scanning products. However for storage (or application) layers where the vast majority of executable code is from legitimate applications, we have implemented GHE with medium sensitivity. With this setting we've been able to maximize detection rates for unknown malware while having a negligible false positive rate.

The results of the GHE so far are excellent. Since its release, roughly one third (cumulative) of the new variants received at PandaLabs from real users' machines have been submitted automatically by the GHE.

One step ahead.

3.4 Collective Intelligence

Today there is over 10 times more malware being distributed than two years ago. The obvious conclusion is that a security solution must detect 10 times more malware to provide adequate protection to users. While a full-fledged HIPS solution raises the bar substantially by detecting and blocking most of these with proactive technologies, it is still possible for unknown malware to slip through its defenses. We need to consider the fact that, while 80% or 90% of proactive effectiveness is relatively speaking an excellent score, in absolute terms it may lead to hundreds or thousands of malware samples being missed over time, since even a small fraction of a large enough number will still be a "big" number.

The Collective Intelligence approach is initially released at the end of 2006 in limited pilots with the objective of being able to reliably detect "10 times more than we are currently detecting with 10 times less effort". Collective Intelligence functions as an online and real-time Security-as-a-Service (SaaS) platform. With over two years of research and development behind it and millions of dollars in investment efforts, it is already paying off by:

1. Benefiting from "community" knowledge to proactively protect others.
2. Automating and enhancing malware collection, classification and remediation.
3. Gaining knowledge on techniques to improve existing technologies.
4. Deploying new generation of security services from the cloud.

3.4.1 Benefiting from Community Knowledge

Traditional security solutions are architected with a PC-centric philosophy. This means that a PC is treated as a single unit in time and any malware detected within that PC is considered separately from the rest of the malware samples detected in millions of other PCs.

Traditional security companies do not have visibility into what PC a particular piece of malware was first seen on. Neither is there visibility of the continuity of that malware's evolution over time in different PCs.

Most importantly, other PCs do not automatically benefit of proactive malware detections on different PCs. They have to wait for the antivirus lab to receive that specific sample, wait for a signature to be created, QA'ed, deployed and protect other users.

Ultimately this results in traditional approaches being too slow to combat today's rapidly moving malware.

One of the main benefits of the Collective Intelligence approach, in addition the effectiveness provided by the automation of the malware remediation life-cycle, is the automatic and real-time benefit it provides to the users of the Collective Intelligence Community.

As soon as a malicious process is detected in a users' PC by the Collective Intelligence servers (whether by system heuristics, emulation, sandboxing or behavioral analysis, etc.) the rest of the users worldwide will automatically benefit in real-time from that specific detection. This results in a close to real-time detection not

only of initial malware outbreaks but also of targeted attacks whose objective is infecting a small number of users to stay below the radar.

3.4.2 Automated Malware Protection Process

One of the biggest barriers to raising the bar of reliable malware detection ratios is the fact that the process of creating a signature against a single sample takes too long in the industry. Each malware sample needs to be sent to the lab by an affected user or fellow researcher, reversed engineered by a lab technician which in turn needs to create a detection signature and disinfection routine for it. These in turn need to be quality-assured, uploaded to production servers, replicated worldwide and finally downloaded and applied by customers.

This entire process is, in most cases, mostly manual and can take up anywhere from minutes, to hours or days or even weeks, depending on the workload of the lab engineers and other factors such as sample priority, prevalence, damage potential, media coverage, etc.

The process can be even delayed much longer when "intelligence" or functionality upgrades to the anti-malware or behavioral engines are involved. It is typical of an anti-malware vendor to upgrade its solutions once or twice a year, as each upgrade has a costly testing and deployment process for corporate customers.

Thanks to the Collective Intelligence infrastructure this entire process of malware collection, classification and remediation can be automated and performed online for

the vast majority of samples. Let's walk through the process from the point of view of a computer who has just been exploited and infected by a malicious code.

3.4.2.1 Automated malware collection

The Collective Intelligence (CI) agent gathers information of processes and memory objects and performs queries against the CI central servers which perform a variety of checks against those.

If certain conditions are met, the suspicious file or parts thereof is automatically uploaded, with the users consent, to the CI servers where it is further processed.

Since processes loaded in memory are not subject to many of the cloaking techniques and "reveal themselves", the agent component does not need to contain a large amount of intelligence and uncloaking routines and can therefore be very light.

Panda has built a vast database of malware samples, which are automatically collected, which in turn provides the CI web-service with a real-time feed of new malware classification entries.

3.4.2.2 Automated malware classification

Server-based processing is not limited by the CPU and memory constraints of personal computers. Therefore scanning routines at the CI servers undergo much more in-depth processing by more sensitive technologies (signature and sensitive

heuristics scanning, emulation, sandboxing, virtualization, white-listing, etc.) to reach a final classification.

It is important to note that the scanning power used at the CI servers is only limited by hardware and bandwidth scaling, unlike a typical scenario at a PC, desktop or server machine. Therefore many of the more resource-intensive proactive techniques which PandaLabs is using, and which provide much higher detection rates (at an also higher computational costs) can now be used massively for the benefit of the users without even touching valuable customer's CPU and memory resources.

With this approach the majority of new malware samples can be analyzed and classified automatically in a matter of minutes. The CI servers are managed by PandaLabs and therefore samples that cannot be classified automatically are ultimately looked at by an analyst at the lab.

3.4.2.3 Automated malware remediation

The remediation module of the CI is in charge of automatically creating detection and disinfection signatures for the samples previously analyzed by the processing and classification module. These signatures are in turn used by the community of CI users to proactively detect and disinfect new or even targeted attacks with very low numbers of infected hosts.

The traditional anti-malware and HIPS solutions have also started to benefit from the CI approach. During the initial 3 months of operation the remediation module has created protection for a few hundreds of thousands of malware samples

which have been gradually deployed to our existing products.

One of the main benefits of the Collective Intelligence approach is that these signatures do not need to be downloaded to each client as they operate from the cloud. This however does not mean that the client machine will not need to maintain updated signatures.

A potential threat to such an approach is the availability of the Collective Intelligence servers. However our approach for integration of the Collective Intelligence technology on current solutions is designed as an additional layer of protection. Therefore under non-availability of the platform for whatever reason, security protection would fall back to the regular HIPS solution which provides well above average protection.

3.4.3 Gaining Knowledge on Malware Techniques

Other one of the main benefits provided by the community feature of Collective Intelligence is that of giving insight to our engineers of new malware techniques and distribution points. Questions such as where was a specific piece of malware first found and how did it spread allow us to model additional intelligence into specific malware families and even creators of specific malware variants.

This approach of applying data warehousing and data mining techniques to malware detections by the community provides significant knowledge on how malware and targeted attacks are carried out. The type of knowledge that can be gathered using this approach becomes

especially useful if it can be applied for tracking infection origins, which in turn might have some interesting applications and benefits for law enforcement efforts.

3.4.4 Deploying Security Services "from-the-cloud"

We have developed and deployed a few services already that function purely based on the Collective Intelligence platform. These online services are designed to perform in-depth audits of machines and detect malware not detected by the installed security solution.

For consumers and stand-alone PCs we have deployed NanoScan (www.nanoscan.com) which scans a PC for malware actively running and TotalScan (www.pandasecurity.com/totalscan) which performs a full system scan of the entire PC, including hard drive, memory, email databases, etc.

On the corporate front the requirements for performing and in-depth malware audit are more demanding. Therefore we have created a specific managed service called Malware Radar (www.malwareradar.com). Thanks to this service companies can quickly perform complete audits of their entire network endpoints to verify their level of security, pinpoint non-detected infection sources or to unveil machines which have been subject to targeted attacks.

3.4.5 A note on white-listing

Since 2004 there have been some new companies spawn from under the rocks promising to "*get rid of the virus problem forever*" with a white-listing approach.

Initially this approach might have seemed as an interesting idea back then. However the challenges presented by a white-listing solution to completely prevent malware are varied. Some of the main shortcomings of a "white-listing only" approach are:

1. There are billions of goodware files vs. the few millions of malware files in existence today. For white-listing to be effective you would have to analyze many more files than malware.

2. Every time a new file has to be added to the white-list, it needs to be analyzed to make sure it is not malicious. Simply adding files to the white-list without analyzing them completely defeats the purpose of a white-list. Otherwise how do you prevent malware from being included on the white-list?

3. Every time a new update or upgrade is made available as a Service Pack or Hotfix for Microsoft Windows, Office, QuickTime, Adobe, Java, etc. the white-listed files need to be re-analyzed and re-created.

4. Managing these white-lists on each computer on a network is a manual and tedious job which needs to be done and which network administrators need to find time to perform.

5. If antivirus laboratories who have hundreds of engineering resources cannot keep up with the pace of analyzing all the malware, how much investment in capital and resources should a white-listing company require in order to keep up with the pace of analyzing 100 times more goodware?

6. Anti-malware updates are delivered to customers via signature databases which

are already big in size. However white-listing updates will be much bigger in size. How shall those be delivered to the desktop and companies?

7. What happens when there's new or updated applications that a user or company needs to run which are not included in the white-list? Who will be doing the reverse engineering and analysis of the supposedly benign program and associated files to determine that they are truly non-malicious?

8. What happens when a virus or worm manages to infect files of a white-listed reputable software company's installer package? It has happened in the past a few times already.

Relying exclusively on white-listing technologies might make sense in certain locked down environments such as call centers, ATM machines and the like. But in the vast majority of corporate environments this is not the case.

There have been very active and lively discussions[25][26] lately on the pros and cons of white-listing, specially promoted by the white-listing companies themselves that feed on the "Antivirus is Dead, White-Listing is the solution" rumor.

However the white-listing approach should not be dismissed altogether. It does bring up many interesting opportunities in the fight against malware, but we believe the benefits are much more effective when combined with black-listing and other proactive approaches.

As we have seen during the explanation of the Collective Intelligence platform, a

white-listing component is an important aspect for complementing and improving black-list detection and, specially, reducing false positives and processing times.

One step ahead.

4. Conclusion

The latest advances by the black hat and cybercrime communities are taking advantage of the inherent weaknesses in the security industry: (a) the labs are being swamped by more malware which is being created every day, (b) by remaining invisible users do not perceive the need for additional protection, (c) targeted attacks that only infect very few users are more effective than epidemic attacks that infect millions of users and (d) users tend to trust a single solution or single layer of protection as their main line of defense against malware.

As malware techniques advance in this cat-and-mouse game, security vendors need to add more layers of protection to keep customers safe. The need for additional protection is revealed by the fact that a large portion of users with current and updated security solutions is in fact infected.

To tackle today's problem we need new layers of protection that take advantage of automating the entire malware protection cycle, from sample collection, analysis, classification to remediation. But automation by itself is not enough. We also need visibility into what's happening on all PCs in order to detect targeted attacks more efficiently and gain a competitive edge on malware creators.

The approach developed by Panda Security, called Collective Intelligence, provides all the benefits of an added layer of defense that provides effective response and protection to the current malware threats, is able to detect targeted attacks and gains intelligence thanks to the correlation of all the detections by the community of users.

5. References

1. Research Study: Active Infections in Systems Protected by Updated AntiMalware Solutions. Panda Research. August 2007.
http://research.pandasecurity.com

2. Gartner's 10 Key Predictions for 2007. Gartner. December 2006.
http://www.eweek.com/article2/0,1895,2072416,00.asp

3. The Zero-Day Dilemma. Security IT Hub. January 2007.
http://www.security.ithub.com/article/The+ZeroDay+Dilemma/199418_1.aspx

4. Welcome to 2007: the year of professional organized malware development. F-Prot's Michael St. Neitzel at Hispasec. February 2007.
http://blog.hispasec.com/virustotal/16

5. Call the cops: We're not winning against cybercriminals. ComputerWorld. February 2007.
http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9010041

6. The Long Tail: malware's business model. Panda Research. January 2007.
http://research.pandasoftware.com/blogs/research/archive/2007/01/08/The-Long-Tail_3A00_-malware_2700_s-business-model.aspx

7. List of online scanners. CastleCops Wiki.
http://wiki.castlecops.com/Online_antivirus_scans

8. Kernel Malware. F-Secure. February 2007.
http://www.f-secure.com/weblog/archives/archive-022007.html#00001118

9. Antirootkit.com List of Rootkit Detection & Removal Software.
http://www.antirootkit.com/software/index.htm

10. Rootkit used in Vodafone Phone Tapping Affair. July 2007.
http://www.antirootkit.com/blog/2007/07/12/rootkit-used-in-vodafone-phone-tapping-affair/

11. Panda Anti-Rootkit. April 2007.
http://research.pandasoftware.com/blogs/research/archive/2007/04/27/New-Panda-Anti_2D00_Rootkit-_2D00_-Version-1.07.aspx

12. Packing a punch. Panda Research. February 2007.
http://research.pandasoftware.com/blogs/research/archive/2007/02/12/Packing-a-punch.aspx

13. AV performance statistics. OITC & MIRT. Real-time feed of antivirus zero-day detection.
http://winnow.oitc.com/avcentral.html

14. Attack of the Zombie Computers Is Growing Threat. The New York Times. January 2007.
http://www.nytimes.com/2007/01/07/technology/07net.html?ex=1325826000&en=cd1e2d4c0cd20448&ei=5090

15. 30 Days of Bots Inside the Perimeter. Support Intelligence. March-April 2007.
http://blog.support-intelligence.com

16. Web-Attacker Exposed. Websense. November 2006.
http://www.websense.com/securitylabs/blog/blog.php?BlogID=94

17. MPack Uncovered, May 2007.
http://blogs.pandasoftware.com/blogs/images/PandaLabs/2007/05/11/MPack.pdf

18. The World's Smallest Downloader. Symantec. December 2006.
http://www.symantec.com/enterprise/security_response/weblog/2006/12/worlds_smallest_downloader.html

19. Packing a punch (II). Panda Research. March 2007.
http://research.pandasoftware.com/blogs/research/archive/2007/03/20/Packing-a-Punch-_2800_III_2900_.aspx

20. Banking Targeted Attack Techniques. Panda Research. March 2007.
http://research.pandasoftware.com/blogs/images/Panda-eCrime2007.pdf

21. Host-Based Intrusion Prevention Systems (HIPS) Update: Why Antivirus and Personal Firewall Technologies Aren't Enough. Gartner. January 2007.
http://www.gartner.com/teleconferences/attributes/attr_165281_115.pdf

22. A Very Large Honeynet. Panda Research. December 2006.
http://research.pandasoftware.com/blogs/research/archive/2006/12/19/A-very-large-malware-honeynet.aspx

23. Panda TruPrevent Personal 2005. PC Magazine USA. November 2004.
http://www.pcmag.com/article2/0,1759,1727653,00.asp

24. The Last Great Security Crisis. eWeek February 2007.
http://www.eweek.com/article2/0,1895,2095118,00.asp

25. Comments on "The Decline of Antivirus and the Rise of White-Listing". The Register. June 2007.
http://www.theregister.co.uk/2007/06/27/whitelisting_v_antivirus/comments/

26. "More on White-listing". Kurt Wismer. June 2007.
http://anti-virus rants.blogspot.com/2007/06/more-on-whitelisting.html

PANDA SECURITY | One step ahead.

PANDA | One step ahead.

PANDA SECURITY

Panda SPAIN
Ronda de Poniente, 17
28760. Tres Cantos. Madrid. SPAIN
Phone: +34 91 806 37 00

Panda USA
230 N. Maryland, Suite 303
P.O. Box10578. Glendale, CA 91209 - USA
Phone: +1 (818) 5436 901

www.pandasecurity.com

PANDA | *One step ahead.*
SECURITY