available at www.sciencedirect.com

ScienceDirect

ELSEVIER

Information Security Technical Report

# Cloud security technologies

*Igor Muttik*[*], *Chris Barton*

*McAfee® Avert® Labs, Alton House, Gatehouse Way, Aylesbury, Herts HP19 8YD, UK*

ABSTRACT

You may have heard a new term that started making rounds very recently – ''cloud-based security''. In this paper we describe past and contemporary security technologies based on the knowledge provided from the servers in the Internet ''cloud''. We discuss how cloud-based malware scanners can simbiotically coexist with traditional scanning technologies, what are the advantages and limitations of the new approach. We also touch on the privacy aspects and challenges related to testing (especially comparative testing) of the cloud security solutions.

## 1. Introduction

Personal computers, networks, and the Internet have been around for fewer than 25 years, but they have already deeply penetrated our everyday life. A computer that has no network or Internet connection is relatively useless for most of our day-to-day tasks. At the same time, however, the face of computer security has also changed: threats that used to come from floppy disks now increasingly come from the network and the Internet.

The cornerstone of traditional computer security technology has always been anti-virus (anti-malware) software—a standalone, updateable application that can determine whether another program is clean or malicious. Since the very first anti-malware programs appeared in the 1980s, they have changed little in how they work. But in the face of dramatic changes in network technology, computer security must also adapt.

In this paper we shall look at how computer security software can protect from malware whilst using knowledge stored on servers in the Internet ''cloud.''

By cloud security technologies, we mean any kind of computer protection that actively communicates with external servers (usually Internet based). This communication can, for example, provide feedback from online-detection databases, reputation systems, black- and whitelists, managed services, and so on. This rapid feedback can give security software the necessary edge it needs to thwart malicious behaviour.

## 2. Anti-spam and Web filtering relies on online checks

For many years virtually all anti-spam products have queried online servers for the latest anti-spam signatures before scanning. Techniques include using blacklists like SURBL [http://www.surbl.org/] and URIBL [http://www.uribl.com/] to catch links in the body of messages (because 70–90 percent of emails contain a link to a site). Checking a site's IP reputation was provided by Spamhaus [http://www.spamhaus.org/] and CBL [http://cbl.abuseat.org], and was employing Razor [http://en.wikipedia.org/wiki/Vipul's_Razor] and Pyzor [http://pyzor.sourceforge.net/] to check a handful of ''fuzzy hashes'' for combinations of attributes. For many years McAfee has hosted a significant proportion of these services across 3 diverse data centres as a good-will gesture to the community. The individual queries to the blacklists are tiny, just about 120 bytes; however, our systems see many millions of queries every hour of the day.

The reason for making clients check online databases is that distributing these large updates to hundreds of millions

[*] *Corresponding author.*
E-mail addresses: igor_muttik@avertlabs.com (I. Muttik), chris_barton@avertlabs.com (C. Barton).

1363-4127/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.istr.2009.03.001

of desktops every few minutes is virtually impossible. In addition to the difficulty of reaching so many clients, most machines running scans probably need only a tiny percentage of the updates to be effective.

Online anti-spam solutions have had the power to blacklist an email for about 7 years! But not only anti-spam solutions rely heavily on the ''cloud'' technology. Another example is Web filtering and reputation services like SiteAdvisor [www.siteadvisor.com] which is one of the most popular web browser protection plug-ins. SiteAdvisor alerts users to malicious, inappropriate, or dangerous websites, instant messenger links, and search results. The big difference between SiteAdvisor and SURBL/URIBL is that the former uses an HTTPS connection to the servers because the amount of information sent to clients is far greater than the ''yes/no/maybe'' results of remote anti-spam blacklists. With SiteAdvisor the servers can return to clients a lot more data, covering the site's popularity and prevalence, country, and even site's relationship to phishing, malicious software (malware), and drive by exploits, amongst others.

Mentioned anti-spam and Web filtering systems have had their share of growing pains, but successfully managed to stay a few steps ahead of the bad guys. We've countered every challenge thrown at us—from simple drive failure and database curiosities to the distributed denial-of-service (DDOS) attacks of last year—thanks to an infrastructure replicated across multiple sites and contemporary failover mechanisms.

Given the success with the anti-spam and Web reputation efforts, why not take the approach against malware? After all, most malware these days are Trojans, and they have a pretty short lifetime, very similar to the way spammers exploit the domains they spam and the IP addresses of bots they herd.

## 3. Anti-virus and cloud technology

The following graph shows the number of threats over the last 3 years (note that the threat figures are NOT cumulative) (Fig. 1).

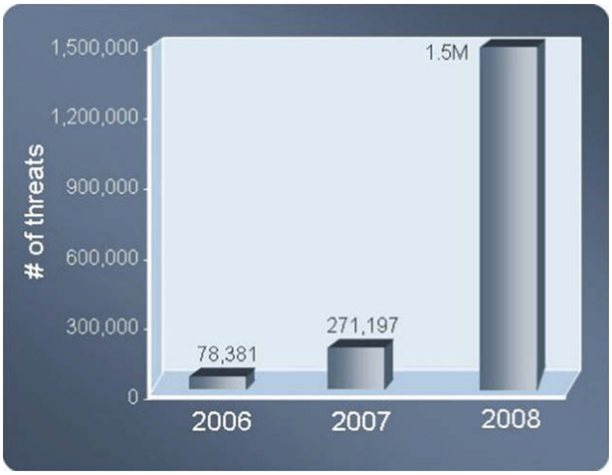

**Fig. 1 – Year-to-year comparison of the number of threats (non-cumulative).**

After 2003, malware attacks, fuelled primarily by financial incentives, grew considerably in frequency. That means, of course, that protection updates also needed to be delivered more frequently. Thus anti-malware companies now release their updates more often. But logically we have to conclude that the best protection is the type that is delivered instantly. Contemporary networks are getting quicker and quicker so it is now possible to perform anti-malware checks without disrupting normal computer operations and to achieve almost instant delivery.

For traditional anti-malware solutions the timeline between when a piece of malware is discovered and the point when protection is available looks like as shown in Fig. 2.

It takes 4 steps to protect the endpoint from a piece of malware and, of course, delays could potentially occur at every one of these steps. That's just not quick enough. Cloud-based technology allows a much speedier reaction (Fig. 3).

The quicker deployment of protection clearly offers better security for all users. But, apart from quicker deployment of protection, cloud technology can solve another problem—the rapidly growing footprint of anti-malware software that causes increasing delays for computer operations.

From the malware growth curve above we can conclude that protection would require growing disk and memory footprints. At the same time malware becomes more short-lived (like spam). Increase in software footprints results in slower computer operation and more bandwidth to deliver updates is consumed. Constantly increasing uniqueness and shorter lifetime of malware also mean that most of the functionality in these updates is never used because each computer does not encounter the great majority of threats.

One solution to these problems is to keep the latest protection in one place, in the cloud servers, which the client security software can check whenever necessary. This solution can somewhat limit the rapid growth of local anti-malware footprints that we are currently observing. At some point in future the footprints might even start declining due to most protection content moving into the cloud.

### 3.1. *Slow-moving clouds?*

It is relatively easy for anti-spam software to perform an online lookup of, say, a domain that appears as a link in an email. A slight delay in email delivery is not going to disrupt normal computer use because the lookup happens before a piece of email is delivered and emails are not expected to be delivered instantly. A delay of up to a few seconds will not generally be a problem. Anti-malware scanners are in a much more delicate situation: If a user is running an application, any noticeable delay will decrease productivity. If a security technology based on the cloud approach will introduce noticeable delays, it will not be used. It's as simple as that.

At the same time, if a cloud technology cannot provide speeds comparable to that of local anti-malware scanning, then it would have a hard time competing with traditional anti-malware software. If, for example, an anti-malware scan of a program takes 50 ms, then it would be possible to match this sort of speed with a rapid cloud check (assuming the local network allows a reasonable speed of operation).

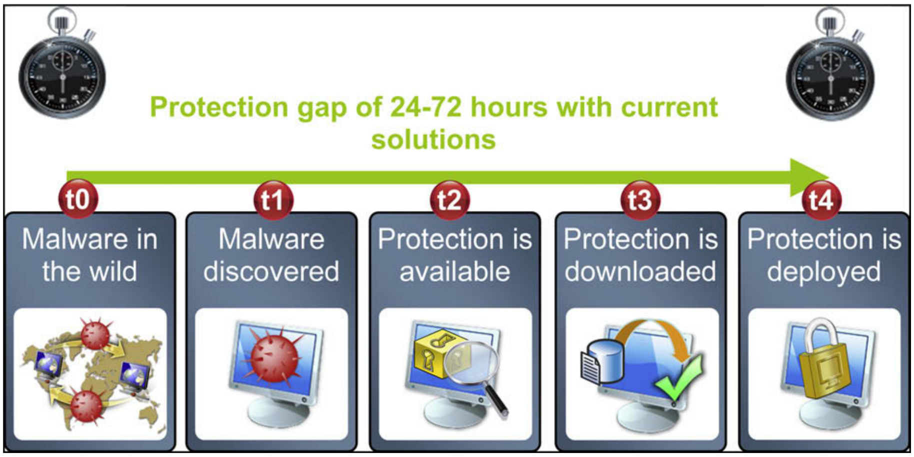

**Fig. 2 – Traditional steps from malware discovery to when protection is deployed.**

Fortunately, many mitigation strategies can be applied to avoid delays:

- Whitelist exclusions,
- Cloud scanning applied in layers,
- Delays related to cloud lookups can be offset by the reduced growth of the disk/memory footprints of security software.

Let us discuss these points in greater detail.

A logical way to introduce cloud-based security protection is to merge it with the existing anti-malware software. There are huge benefits to this approach. Anti-malware has existing infrastructure to deliver updates to all protected servers and desktops. This mechanism is not as quick as we need it to be to counter fast-moving malware, but the anti-malware updates can be used very successfully to deliver baseline protection from common and persistent threats like viruses and worms. Moreover, these updates to the cloud client can also deliver rules and whitelists ensuring that network lookups will not occur for common clean files. A simple updateable whitelist filter can reduce the necessary level of network lookups by several orders of magnitude. So traditional anti-malware updating can be harnessed to deliver updates to the cloud technology—allowing quick and seamless adoption of the new technology because the necessary infrastructure is already in place.

Another benefit from blowing the cloud into anti-malware scanners is that anti-malware is already capable of checking incoming objects on many levels. For example, scanners have all the necessary functions to check email messages and file attachments, URLs, objects downloaded from the network or the Internet, files on removable media, and programs created by network services and applications. It is essential to check programs coming into the system via any of these routes; if cloud checks are performed at all these entry points then the delays introduced by additional network lookups can be reduced. The reduction comes from the fact that any computer system holds and uses many thousands of programs on a daily basis while only very few new ones are usually added (excluding, of course, exceptional cases like, for example, deployment of software service packs or massive software installations; and even in these cases there are usually very few incoming installers and setup packages that require thorough verification).

A network lookup, for example, to check all programs coming in as email attachments is not going to be noticeable. Firstly, such attachments are not common and, secondly, anti-malware can do such checking in the

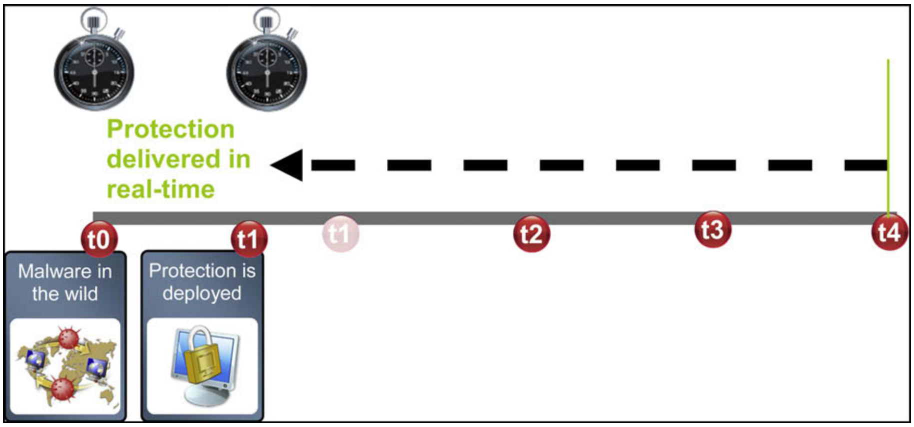

**Fig. 3 – Cloud-based system allows to reduce the delay in deploying protection.**

background (on delivery), before a user looks at an email. Similarly, adding a cloud lookup for every download will not be noticeable. A few extra network packets may add 1–2 KB of network traffic per each downloaded object, but given the average size of contemporary downloads this is unlikely to ever be noticed.

To keep delays down, the cloud technology must be designed with speed in mind. The number, size, and frequency of transmitted and received packets should be as low as possible. To support such high-speed operations, the cloud servers should also be quick and responsive with no timeouts, low latency, necessary redundancy, failover mechanisms and appropriate geographical distribution.

### 3.2. *Offline coverage?*

Merging anti-malware at the endpoint with cloud technology has another benefit: If the latter does not work (for example, when a computer is disconnected from the Internet), then protection can fall back onto traditional anti-malware. Because most malware threats today come through the network, the loss of connectivity to the cloud actually reduces risk.

Anti-malware and cloud technology can operate as an effective symbiosis. Anti-malware will provide long-term protection from persistent threats (for example, from replicating malware such as parasitic viruses and worms) while the cloud part will provide rapid protection from short-lived bursts of other malware.

An interesting thing about being ‘‘offline’’ is that some Internet protocols really do operate even before you are, strictly speaking, connected to the Internet. For example, when you have a physical connection in place (when the network cable is connected and you can see the LED on the Ethernet card blinking, then you normally have an IP address assigned via DHCP) you may still need to pay a fee to browse the Internet. But at that point DNS is already working because it’s DNS that will redirect you to ‘‘here is what the fee is’’ web page. Thus your cloud security (if it is DNS based) can be active before you actually pay and start browsing!

## 4. Bandwidth and latency

A ‘‘chatty’’ client in a cloud-based security solution can eat a lot of bandwidth on the LAN. Such a client would also likely cause more stress on the servers, which in turn would diminish the overall speed of the replies to other clients. Thus it is best to minimize the size and frequency of the communication between client and cloud server. Lightweight protocols have an advantage—UDP instead of HTTP, which requires a handshake before any transmission to establish a connection, or HTTP instead of HTTPS, which must exchange security information and that, of course, adds to the overhead.

To reduce the bandwidth requirements we can employ several approaches:

- The cloud solution can be merged with traditional anti-malware, which can filter all known malware and thus reduce the number of network lookups.
- The cloud client can use a local whitelist to avoid performing checks of known good files.
- Caching replies can spare repeated queries (at least for a certain time).

Latency is especially critical to any cloud technology. Geographically distributed servers should allow a network to meet reasonable targets on query latency. As a rule of thumb, cloud queries should take no longer than a thorough scan of a file on disk.

Latency may increase when a server is overloaded. This leads to another risk to cloud technology—an orchestrated DDOS attack on the servers. It’s logical to expect that the bad guys will eventually try to circumvent cloud technology by the simplest means already available to them—by commanding their botnets to launch a DDOS assault. Fortunately, we can take measures against such attacks, particularly by filtering unwanted traffic or using a distributed network of servers (for example, Akamai). To some extent, the geographical distribution of servers to achieve low latency levels already greatly increases the resilience of the cloud to such attacks.

A beneficial side effect of merging anti-malware with the cloud is that the size of anti-malware updates will likely decrease once this method is widely adopted because the protection by the cloud part will make it unnecessary to roll out some definitions to all clients (especially to protect against malware that was not found in the field or was only seen once).

Apart from minimizing bandwidth consumption and maximizing the speed of operation, we must make sure that communications are secure.

## 5. Security

The ‘‘cloudness’’ of a security solution opens some opportunities for attackers to manipulate the normal operation of the system. An attacker can:

- Intercept the data sent from the client to the server,
- Intercept and/or spoof the reply from the server to the client,
- Launch a DOS attack on the server.

Intercepting outgoing client data (with a sniffing tool, for example) is very unlikely to have privacy implications because transmitted information (as long as a security product is following current legislation and accepted standards) is not going to contain sensitive data. Sometimes, however, the mere fact that such a transmission occurs may be considered important (for example, as an indication whether a system is likely to be compromised). Some measures can be taken to minimize this risk such as using the DNS protocol, in which case all external communication becomes anonymous. (The DNS server, not individual clients, will appear to be the source. The clients’ IP addresses will not be visible externally.)

Protecting the clients from spoofed answers should be a key concern from Day One. If your infrastructure is sound (with appropriate firewalls in place and proper physical security), then so, too, should be the communication between the clients and the cloud.

The worst-case scenario is if the cloud gives a false alarm when for an innocent file a positive (“malware found”) response is generated. The product’s default action would occur; usually delete or quarantine and a message box would pop up with a malware alert. It could be catastrophic if an attacker would be able to forge a detection response for any critical system file (such as *winlogon.exe* or *ntoskernel* ). This scenario can be avoided if the communication from the server is digitally signed. The client can check the signature and discard any invalid replies as spoof attempts. An alternative is to use encryption such as SSL between client and server throughout a TCP/IP session. (That’s what the HTTPS protocol is doing automatically.) This method is slower but addresses the spoofing problem, too.

## 6. Privacy

Cloud scanning can be implemented in many ways; it may or may not have privacy implications. For example, if a cloud client transmits only a hash of an object (or a part of an object) to the server, then there is no way to extract any private information from this hash.

On the other hand, a cloud client designed with a poor understanding of privacy implications can transmit just about anything to the server—the whole scanned object, the folder names, or the IP and MAC addresses of the computer. (The MAC address is a unique number identifying a network card; these addresses do not repeat, so they can be used to identify a specific computer.) Transmitting this information would be an extreme violation of privacy because the files might contain not only personal data but also financial or classified information. And all these data can be linked to a specific computer via its IP and MAC addresses.

Any security provider has to make a balanced choice of which information is transmitted. They must not send anything that can be personally identifiable (such as a MAC address). They should send as little information as possible—the bare minimum necessary for the purpose. This information must also be anonymous.

We also need to underline that the type of communication protocol has a direct bearing on the level of privacy. A DNS transmission, for example, goes from a client to its local DNS server (in the enterprise it will be an internal DNS; for home users it will be the DNS of the ISP), which in turn contacts the server. This shielding by the local DNS server ensures that the cloud server never sees the IP address of the client. All the cloud server can see is the IP address of the DNS server. Many thousands of users can share a DNS server, so personal identification becomes impossible.

When TCP/IP communications (such as HTTP or HTTPS protocols) are used, however, then the server should take care that IP addresses are not misused. Ideally, they should not be logged directly but rather translated into geographic locations, which can be stored instead of a specific IP address. In other words, if the incoming information is not completely anonymous, the server software must immediately cover those tracks.

It is extremely valuable for security purposes to check the file’s path, name, attributes (such as a digital signature and its creator), and similar. This information is almost never personally identifiable but in some rare circumstances it could be. A Word document, for example, with the name “Business proposal—acquisition of Wonka anti-malware by IBM.doc” may be rather sensitive. Fortunately, most contemporary malware is relying on the code that exists in files of executable type (like EXE and DLL). Transmitting the attributes of such objects is safe from the privacy point of view because these programs virtually never contain any sensitive data or carry a revealing file name. At the same time, extreme caution should be taken when attributes extracted from other kinds of objects are sent to the cloud, especially user documents with free content (like DOC, DOCX, PPT, XLS, and PDF).

The cloud client should encrypt any information it sends. If not properly encrypted, transmitted data could be intercepted and examined by attackers. We already mentioned that such an interception could have security implications and assist in spoofing replies or analysing the weaknesses in the network environment. But if sensitive data are transmitted, it might have privacy implications too, revealing, for example, names of people, account names and similar.

All security providers who use cloud technology for security purposes must publish corresponding privacy statements and abide by them.

## 7. Testing challenges

Proper testing of traditional anti-malware solutions is very challenging—and is getting harder all the time due to their growing complexity. Testing cloud technology or a combination of anti-malware and cloud-based protection is an even more complicated task.

Tests are usually run with “frozen” products and sample sets, so that the results are reproducible, repeatable, and verifiable. Unfortunately, you cannot use the same approach with a cloud-based security solution because it cannot be frozen. The server side is always going to be fluid and, being outside of tester’s control, not “freezable.” Consequently, the test environment becomes fluid, too. Results cannot be rerun and the chances of finding any mistakes in the tests are greatly reduced. From the tester’s point of view, this is equivalent to introducing a “black box” into a test, and the worst thing is that black box may contain humans and third parties (for example, ISPs). On the other hand, the developers of tested products may view such tests as unreliable, and thus suspicious, because nothing can be reproduced.

Tests of cloud-based anti-malware solutions are only just starting so there are no universally accepted methodologies yet. The main idea of carrying out such tests is to run security products over a long time, from different IP ranges and geographies. The average results of such testing can be obtained periodically. (For example, the average latency of a reply and the average detection rate during the last month could be published everyday.) Such a statistical approach should level out short-term fluctuations related to the timing of publishing specific protection rules, delays at the ISP level, the temporary load on servers, etc.

Competitive testing of cloud technologies adds an additional level of complexity because multiple products should be compared, with each of them being ''fluid.'' The timing of the tests becomes crucial as well. Because protection can be delivered almost instantly, any product tested earlier will have an unfair advantage. That dictates a need to run tests in parallel.

If you are interested in practical advice about comparative testing of cloud-based products, the Anti-Malware Testing Standards Organization (AMTSO) is currently working on a document describing the best practices for such tests. This document is expected to be adopted by AMTSO in the first or second quarter of 2009 and will be published at www.amtso.org.

We urge all those interested in learning more about this subject to read the whole AMTSO document once it is published. It will contain many more details and practical-use cases.

## 8. Conclusion

By using cloud technologies to reduce the ''exposure'' gap (from days or hours to just seconds), networks and clients can achieve far superior protection. Security products could also become more ''lightweight,'' as they would not need to include and update millions of definition records—most of which are never used. We foresee cloud-based security as a mainstream security technology and predict that it will dominate the field in just a few years. There are risks, primarily to privacy, in cloud-based security; but those should be manageable. The overall benefits of this new system are too great to pass up.

## Acknowledgements

We are very grateful to our McAfee Avert Labs colleague Alex Hinchliffe for valuable discussions and to the AMTSO members with whom we had the pleasure of working on the best practices for our ''cloud testing'' document: Luis Corrons, Alice Decker, Dmitry Gryaznov, John Hawes, Gunnar Johansen, Rick Moy, Denis Nazarov, Michael Parsons, Maksym Schipka, and Mika Stahlberg.

**Igor Muttik** graduated from Moscow State University in 1985. His Ph.D. in 1989 was based on the research of semi- and super-conductors. He became interested in computer viruses in 1987 when PCs in the lab were infected with Cascade.

In 1995 he joined Dr. Solomon's Software in the UK as a Virus Researcher. In 1999 he headed Avert in Europe and now is Senior Avert Labs Architect. He speaks regularly at security conferences.

Igor lives in England with his wife Elena and 3 children.

**Chris Barton** has spent a third of his life with McAfee as a research scientist, specializing in new technology and internet physics. He regularly blogs, speaks at specialist conferences and contributes some of his spare time to projects that make the internet a safer place. Chris is the pioneer responsible for the cloud, it's at his house. Chris lives on the Internet and is one of the 'voices' behind McAfeeAvertLabs on Twitter.