

## WHY CONVENTIONAL ANTI-VIRUS TECHNIQUES WON'T STOP NEW THREATS

Vipul Ved Prakash, Founder and Chief Scientist, Cloudmark, Inc.  
Adam J. O'Donnell, Senior Research Scientist, Cloudmark, Inc.

December 2005

### ABSTRACT

Viruses and spam are both threats to productivity, but the techniques developed to combat them, like the abuse itself, differs radically. Consider that the skills required to create and generate a virus message are advanced, while the skills required to generate spam are minimal — which is why there's so much of it. Conventional anti-virus software is created by company researchers who must first isolate a virus and then generate a fingerprint of the virus that can be checked against a database of known viruses. This methodology is effective only at a small scale and with infrequent variations in viruses. Spam mutation, on the other hand, appears more frequently, almost constantly. Since it would be infeasible to update anti-spam software manually with every variation, it must be automated. Conventional anti-spam software looks at patterns, such as words found in the message, or the route the message took. Cloudmark, however, uses a collaborative filtering-based, anti-spam solution, at the heart of which is the Cloudmark Collaborative Security Network™. This solution relies on a large pool of email readers to distinguish what is, and is not, spam. All messages are reduced to a fingerprint, and once enough readers identify a message as spam, that fingerprint is labeled as spam. All messages associated with that fingerprint, as well as specific variants of the message, are filtered into the spam folder. This technique not only works for spam, but has also proven to be extremely effective against viruses, phishing, and spyware — and at real-world speeds much faster than conventional anti-virus products.

Spam and viruses are both unwelcome intruders on our computers. They consume the time and energy that we bring to the workplace. Everyone recognizes that a virus can sap productive work hours during a major outbreak. Spam, on the other hand, is rarely a one-time, large event. Instead, spam tends to consume a small amount of an employee's time every day. In effect, viruses are malicious. They threaten data integrity and overload systems quickly, often drastically, destroying productivity, while in contrast, spam is a death by a thousand cuts. Entire industries have developed to combat these two security threats, including the messaging anti-abuse industry of which the authors are members. Currently, however, the techniques that have been developed to combat viruses and spam differ radically from one another.

Conventional anti-virus software works on a basic principle. The software examines every file that comes into the machine and generates a unique fingerprint for each file, which is then checked against a database of fingerprints of known viruses. Typically, the fingerprints are created by company researchers who isolate individual samples of computer viruses and create a mathematical representation that is unique to only that piece of code. If the database is updated frequently, then all viruses are filtered out of the system before they can do any harm.

Conventional anti-spam software works on a completely different principle. The algorithms used by these software packages exploit characteristics found in spam messages that are not present in standard mail. Probably most common are rules-based systems that look for patterns, such as words used in the message, or the route the message took. Other spam filters exist at the edge of the network and employ rate-limiting techniques that simply kill connections attempting a high rate of mail delivery that is typical of a mass mailing. Another technique is to block connections of mail servers known to be predominantly used to pass spam, rather than legitimate e-mail.

Why have the messaging security approaches to combat viruses been so different from the technology applied to spam? The differences between software architectures are due to the fact that the factors that drive the creation and mutation of spam are radically different from those that foster viruses. While both viruses and spam are unsolicited and unwanted content that arrive at, and sometimes overwhelm, our computers, spam and viruses exist in different ecologies.

First, many more people can write spam than can write viruses. Virus writers are extremely computer proficient as compared to the general populace. Even altering an existing virus to avoid anti-virus filters takes a great deal of ability in comparison. By contrast, the skills required to create a spam message are the same as those needed to write an e-mail or create a graphic. The skills required to generate spam are minimal—which is why there is so much of it. Second, many more people can identify a spam message than can recognize a virus. The average person can quickly differentiate legitimate messages from loved ones and coworkers from illegitimate messages from mass-marketers.

Viruses are more difficult for the average user to recognize. A virus may come across a network silently or by trying to present itself as a “normal” message. The factors that drive virus creation, as opposed to spam creation, also require anti-spam software to support features that are not usually found in conventional anti-virus solutions. For example, due to the far larger number of spam writers, as compared to virus writers, spam mutations appear far more frequently than virus mutations. To effectively combat the sheer amount of spam mutation, anti-spam software must be updated far more often than anti-virus software. The updates necessary to effectively block spam need to occur with such frequency that it makes it economically infeasible to accomplish with a centralized team of “experts.” (In fact, even the majority of conventional “rules-based” anti-spam software fails in this area.) Identifying and filtering variant spam and viruses quickly enough to stop their proliferation is key to an effective system.

To do this, we can use this large pool of e-mail readers to differentiate between spam and legitimate mail. This approach virtually eliminates the cost, time, and human resource factors that currently limit frequent updates. This allows for the creation of a collaborative filtering-based, anti-spam solution—where the masses of e-mail users can, as a group, decide on the nature of individual messages by nominating messages as either spam or “not-spam”. In other words, anti-spam techniques that leverage the human-based “Wisdom of the Crowd” can quickly and efficiently determine the spam disposition of a message, use these decisions to generate fingerprints of the spam messages, and then redistribute these fingerprints to spam filters.

The principle of community-based collaborative filtering was critical in the design of our anti-spam system, known as the Cloudmark Collaborative Security Network<sup>1</sup>. While the design is quite complex, the concepts behind its operation are simple. Users submit fingerprints of messages they believe to be spam to a central system. If enough members of the community agree that the content is spam, then the central system labels the fingerprint as truly representing spam. Each piece of e-mail received can be checked against this list to determine if it is spam or not. Just like in the real-world human community, individuals who behave well by quickly and correctly identifying spam become trusted, and are rewarded by having their opinions weighted more heavily in the future.

If there is anything that the reader should take away from this article, it is that the collaborative filtering architecture is not limited to combating spam. Viruses attached to spam e-mail can be identified by the community as well. Phishing and a whole host of other messaging abuses that are recognizable by the average person can be solved by pooling their opinions together to form a communal opinion. In the end, the technology has the potential to democratize the otherwise autocratic world of security software.

#### AUTHOR BIOS

**Vipul Ved Prakash** is the Founder and Chief Scientist of Cloudmark, Inc. He is best known for creating Vipul's Razor and the Cloudmark Network Classifier. Vipul is a prolific Open Source developer, and has written numerous extensions to the Perl programming language for networking, cryptography and object technology. He is a frequent speaker at industry conferences and academic events on the issues of computing and spam, and has been published in First Monday and Perl.com. In 2003, MIT's Technology Review named him as one of the Top 100 Young Innovators in the World.

**Adam J. O'Donnell** is a senior research scientist at Cloudmark, Inc.. He completed his PhD as a National Science Foundation (NSF) Graduate Research Fellow in Drexel University's Department of Electrical and Computer Engineering. Adam has worked on several technical articles and books, serving as a contributing author to "Building Open Source Network Security Tools" and "Hacker's Challenge", and co-author of "Hacker's Challenge 2".

#### REFERENCES

1. V. V. Prakash and A. O'Donnell. *Fighting spam with reputation systems*. Queue, 3(9):36–41, 2005.