Spam Testing Methodology

Opus One, Inc.
March, 2007

This document describes Opus One's testing methodology for anti-spam products. This methodology has been used, largely unchanged, for four tests published in Network World magazine as well as in other Opus One testing projects.

Methodology Overview

Opus One evaluates anti-spam products by installing them in a production mail stream environment. Each Opus One test includes a minimum of 10,000 messages, and usually covers a period from 7 to 14 days. Each test feeds the same production stream to each product under test simultaneously, recording the verdict (typically "spam," "not spam," or "suspected spam") for later comparison.

All software-based engines are installed within virtual machines (VMware server) on several dual-CPU/quad-core servers specifically dedicated to this task. Where the engine requires a base operating system, Windows 2000 is used except in cases where the product requires a different operating system or operating system version. Each VMware server is configured with 16 Gbytes of physical memory and a combination of RAID 1 and RAID5 SCSI disk arrays to provide best possible disk performance on this platform.

Each product tested is acquired either directly from the software/hardware vendor, or through normal channels (including the secondary, used, market). Each product tested is under a current support contract.

It is believed that each product is "up to date" with publicly released software at the start of the test. Where multiple versions are available from a vendor, the technical support team for each vendor is consulted to determine the most "recommended" platform to use. To minimize confusion, products are not upgraded during the term of the test (although anti-spam and anti-spam engine updates are usually made by each product automatically during the term of the test). All systems are able to connect to the Internet for updates and DNS lookups. A firewall is placed between each product and the Internet to block inbound connections. However, outbound connections are completely unrestricted on all ports.

Each product is configured using the recommended settings from the product manufacturer. These settings are labeled "Out of the Box" (OOB). Where easily done, multiple scenarios are used for a product, including an aggressive setting ("tuned up"), and a conservative setting ("tuned down"). In some cases where obviously inappropriate settings are included by default, these are changed to support the Opus One production mail stream. The most common setting changed is "maximum message size," which often has a very small default, such as 1Mb.

In case of any ambiguity in settings, Opus One invites the technical support team of each product vendor to evaluate the configuration and certify it as appropriate for a typical enterprise anti-spam environment.

Message Flow

Production mail traffic is received from the Internet by the normal Opus One MTA, scanned for viruses, and immediately re-transmitted to each of the test systems or virtual machines. Each message also has a special X-header added to it (X-Originating-SIP) showing the originating IP address of the message. Each of the reputation services being is also tagged into the message at this point using the original IP address. These are used (where possible) to evaluate the reputation-based service behavior separately from the content-based behavior.

Products that also consider reputation data during their content-based filtering are configured to know either about the X-header with the original IP address, or are given sufficient topology information (in the form of hop count, typically) to extract the original IP address from "Received:" headers. The Opus One MTA is believed to be fully RFC-2822/RFC-2821 compliant with regard to message and header format.

It is important to note that the message stream used in our tests is a real corporate message stream, including both internally sourced and externally sourced email messages. It contains no artificial content and can be considered to represent a normal US-based enterprise stream. No spurious spam or non-spam content is injected into the stream.

Pre-scanning the email for viruses is an important part of the test. Because nearly 100% of the virus-infected email is actually created by mass-mailing worms (rather than true messages from people with infected attachments), many anti-spam products are beginning to treat mass-mail worms as "spam." We don't want the complication of dealing with differing interpretations of certain kinds of viruses as spam, so we pull them out of the mail stream where possible.

Once we clean the incoming mail stream of viruses, we turn it around and simultaneously re-feed it to each of the VMware systems and appliances in very close to real time. Typical delay between message receipt and re-transmission is less than 1 second. From the point of view of anti-spam testing, we can consider that this gives us instantaneous delivery of a contemporaneous real enterprise mail feed to all products simultaneously. An important part of our test is that each of the systems was seeing the mail as close to the time we received it as possible. In different tests conducted by Opus One, where we re-fed the same stream to anti-spam products both hours and days later, we turned in dramatically different scores, showing how important spam updates can be for products that depend on signature-based technology.

In the case where a product does not accept a message immediately, the Opus One MTA queues the message and attempts to retransmit it at regular intervals.

Each product is connected to the Internet and was able to get signature and software updates as often as recommended by the vendor. Where vendor technical support teams recommend a shorter update cycle, this recommendation is implemented.

The goal of each test is to get approximately 10,000 messages sent through the test systems.

Once the messages are received, we manually read through every single message, classifying it as "spam," "not spam," or "unknown." We defined as "spam" the

messages for which there is no conceivable business or personal relationship between sender and receiver and which are obviously bulk in nature. In the “not spam” category are mail messages which may or may not have been solicited, but which either had a clear business or personal relationship between sender and receiver, or which are obviously a one-to-one message, even if unsolicited and unwanted. All mailing lists which have legitimate subscriptions are considered “not spam.”

In the unknown category go messages that either are the result of virus double bounces, messages that we could not put into one category or the other definitively based on content, and some messages that were so malformed that we don’t know whether they are spam, viruses, or just software acting up. We also take messages with duplicate message IDs and delete them from the data set. In theory, a duplicate message ID is impossible, but spammers don’t follow the RFCs and we always find a number of these. For example, in one of our tests, a spammer sent the same message ID over 400 times, each with slightly different message content.

Because we delete the "unknown" messages from the data set, all of the statistics are generated without those messages included. In other words, we treat the data as if we got full set of "perfect" messages, rather than include the unknown or unclassifiable messages in our statistics. Thus, all percentages and rates are based on the “spam” and “not spam” message sets and do not include the “unknown” message set.

Once the manual qualification of messages is completed, all results are placed in an SQL database and queries were run to create false positive and false negative lists. Each false positive for each product is individually evaluated and any errors in the original manual classification are fixed. Because the number of false negatives is typically much higher (400 to 700 false negatives per product are not unusual), we do not evaluate every single false negative for each product. Instead, we evaluate every single false negative for at least 2 different products, and we sample the false negative results for all other products to identify any errors in the original classification. This sampling means that some false negatives may be inappropriately marked, although the actual percentages are likely to be significantly less than 1%. Once the data sets are seen to be within acceptable error rates, the databases are reloaded and the queries recreated.

All of the reputation scores are also loaded into the SQL database and used to run queries to determine the results of several "what if" scenarios related to the combination of reputation-based and content-based anti-spam services. Specifically, we look at the absolute block rates for each reputation services, the false positive rates for each service, and we evaluate “how much better” each content-based filter would be under different reputation scenarios.

Message Verdicts

Each anti-spam engine provides a verdict on messages. While this is often internally represented as a number, the verdict in most products is reduced to a categorization of each message as being “spam” or “not spam.” In many anti-spam products, a third category is included, typically called “suspected spam.”

In our testing, we configure products, where possible, to have three verdicts (spam, not spam, and suspected spam). Where products do have three verdicts, this raises a

question of the false positive and false negative rates, because a verdict of "suspected spam" sits in between the two extremes.

For this reason, we normally present statistics for products including the "suspected spam" in two different ways: once as if the "suspected spam" was all considered to be "spam," and then again with the "suspected spam" considered to be "not spam." These are represented in our reports as "MS=S" (i.e., consider suspected spam to be spam) and "MS!=S" (i.e., do **not** consider suspected spam to be spam). For the purposes of comparison, the actual false positive rate and false negative rate for each product should be considered to be somewhere in between these two values (MS=S and MS!=S).

To record the verdict of each product, we use a tag-and-deliver strategy where possible. With tag-and-deliver, we configure each product to deliver to an Opus One SMTP server every non-spam and suspected spam message, but tag a verdict of "suspected spam" in the subject line of the messages as appropriate. In most cases, we configure products to simply drop spam messages rather than tag-and-deliver them. Thus, we treat the absence of a message as an implied verdict of "spam" for that message. The MTA delivering messages to each product also logs each message passing through it. These logs are verified at the end of each run to check that no messages were "lost" in the process of retransmission from Opus One MTA to the products under test.

More on Statistics

The terms "false positive" and "false negative" (along with "true positive" and "true negative") come to us out of the world of diagnostic tests. An anti-spam product is like a test for pregnancy: it eventually comes down to a "yes" or "no" decision (although there are a lot less bodily fluids involved). False positive means the test said "this was spam," and it wasn't. False negative means the test said "this was not spam," and it was.

We often think in terms of error rates, but with many diagnostic tests, the **kind** of error is a big deal---not every kind of error has the same consequences. It's not enough to know that the test is wrong 29% of the time. We want to know what kind of wrong. Spam tests are exactly like that. A false positive means that good mail might have gotten lost, while a false negative is just annoying. We care a lot more about false positives than we do about false negatives. We don't want to just know how many errors there are, but we want to know what type they are as well.

Once you decide that you want to keep false positives and false negatives separately, then you need to stick to your guns. This is what researchers who study other kinds of diagnostic tests do, and this is what you need to do to make the best buying decision. Unfortunately, the path now gets convoluted and a bit confusing.

Four main statistics are used to describe diagnostic tests. "Positive predictive value" (PPV) and "negative predictive value" (NPV) go together. They measure how likely the test is to be correct. Positive predictive value, for example, measures the probability that a message actually is spam, given that the test says that it is. (PPV is computed by dividing the number of true positives by the sum of true positives and false positives.) However, positive predictive value doesn't say how much spam will be filtered out: the number of missed spam doesn't figure into that statistic at all.

"Sensitivity" and "specificity" are the other two statistics, sometimes called the "true positive rate" and "true negative rate." They measure how likely a test is to catch whatever it is testing for. Sensitivity, for example, measures the probability that a message will test as spam, given that it actually is spam. (Sensitivity is computed by dividing the number of true positives by the sum of true positives and false negatives.) Most research on diagnostic tests uses PPV and NPV or sensitivity and specificity to describe how well a test works, because these are well-defined statistics.

We know that if we published statistics called PPV and Specificity that people will simply get confused. So, we tried to figure out what would be most useful in comparing products and predicting their behavior. We boiled it down to two main questions. First is "how much spam will this filter out?" That question is best answered by the sensitivity statistic. It tells us what percentage of the time spam will be identified by the filter. A perfect score would be 100%. For example, in the earlier test done for Network World, there were 8027 spam messages. Barracuda caught 7563 of those, and missed the rest. Forgetting the false positives (because that's a different question), Barracuda gave us a 94% reduction in the spam: 94 out of 100 spam messages are blocked.

The second question is "how accurate is this filter?" Accuracy can be answered by the positive predictive value (PPV) statistic. That tells us what percentage of the time the test filters out mail correctly. Again, a perfect score would be 100%, meaning that when the filter says that something is spam, it's right 100% of the time. Because people like to talk about "false positive rate," we've taken the PPV and subtracted it from 1 to calculate a false positive rate. For example, in the earlier Network World test with 8027 spam messages, Barracuda was wrong 23 times, giving a PPV of .997 or a false positive rate of 0.3%.

Another way that some researchers define false positive rate is by subtracting the specificity from 1 or by dividing false positives by the sum of false positives and true negatives (these end up being the same value). For most products, these numbers are close, although they do measure different things.

Marketing representatives like to create yet another statistic by dividing the number of false positives by the total sample size. One reason for yet another statistic to be created is that this is the smallest possible number they can report---by mixing up false positives, true positives, false negatives, and true negatives in the statistic, the denominator gets big, which means that the result will tend to be small, and it will always be smaller than the accepted PPV and Specificity numbers. There is no statistical term for that number, although you will often see it in white papers and advertising, incorrectly, as "false positive rate."

The nice thing about this vendor-reported number as well is that it sweeps under the rug the fact that a low false positive rate is generally accompanied by a high false negative rate. An example here would be helpful. Let's suppose that 75% of your email is spam. Look at 100 messages, and 75 of them will be junk. Now, let's compare two anti-spam filters. One looks at 100 messages, says that 2 messages are spam, and is wrong about one. The other looks at 100 messages, says that 76 are spam, and is wrong about one. If you use the marketing formula and simply divide the false positive count by 100, both have a 1% false positive rate. But if you use an honest false positive rate, such as the one we use, you can see real differences: the first product has a false positive rate of 50%, while the second one (which is much more accurate) has a false positive rate of 2%.