Введение

Опуская этическую и социальную проблематику, связанную со спамом, мы в данной статье сосредоточимся на способах его доставки, методах обнаружения и подавления. Затем мы обсудим ошибки, часто возникающие при описании этих методов и их возможностей. В заключительной части статьи будет рассказано о «супершингле Яндекса» – несложном техническом методе детектирования массовых рассылок, используемом в Яндекс.Почте [http://mail.yandex.ru].

Часть 1. Доставка спама. Эволюция

Чтобы спам попал в ваш ящик, его необходимо вам доставить. Поскольку мы не встречали внятной классификации спама по способу доставки (а остальные классификации сводятся к простой дихотомии: спам модифицирующий и спам немодифицирующий текст заказчика) постараемся вкратце описать их здесь. Возможно, этот текст покажется излишне подробным, однако нам кажется интересным проследить, как разработчики спамерского ПО откликались на вызов, брошенный им антиспамом, и vice versa.

Спам молод. Как средство активного маркетинга он возник примерно в 1997 году. О дате его возникновения можно судить по моменту, когда Paul Vixie создал RBL. RBL – исторически первая серьезная попытка борьбы со спамом. См. http://www.wikipedia.org/wiki/DNSBL.

Эволюция технических видов спама на 100 процентов обусловлена эволюцией антиспамовых средств. Причем история тут развивается стремительно, по нарастающей. За последние два года в ней, по-видимому, произошло больше событий, чем за все предшествующие.

Прямые рассылки и открытые релеи.

Первые виды спама были просто прямыми рассылками. Такой спам блокируется достаточно просто, и спамеры начали использовать открытые почтовые релеи, то есть обычные почтовые серверы, позволяющие произвольному пользователю воспользоваться сервисом отправки письма на другой сервер. Заметим, что иных релеев в ту пору просто не было, а само понятие «открытые релеи» возникло лишь после того, как появился спам и их вообще начали закрывать.

Такие открытые релеи достаточно легко детектировать, их стали активно искать и блокировать. После этого у прямых рассылок наступил ренессанс – спам стал рассылаться с «диалапов» и для его блокирования системным администраторам пришлось разузнавать и блокировать IP модемных пулов основных провайдеров.

Прокси-серверы. Socks и HTTP

Чуть более 2 лет назад появились как заметное явление более изощренные способы использования чужих, неаккуратно сконфигурированных серверов.

Socks-прокси серверы предназначены для сведения всего интернет-трафика небольших компаний к одной единственной машине, имеющей доступ в интернет. Для работы они обычно используют порт 1080. Если машина допускает неавторизованное соединение с произвольного IP-адреса (типичная ситуация в эру до спама), ее могут использовать

спамеры и для направления своего SMTP-трафика. Интересно отметить, что логи использования socks-серверов обычно не ведутся, поэтому отслеживание истинных источников рассылки даже самими администраторами socks-серверов чаще всего невозможно.

Почти сразу же обнаружилось, что и стандартные открытые HTTP-прокси (типичные порты 3128, 8080 и т.д.), поддерживающие метод CONNECT, легко использовать для того же самого, достаточно в команде CONNECT указать не только имя серверы, но и задать 25-й почтовый порт. Даже любимый всеми «народный» вебсервер Apache, собранный с модулем mod_proxy и неправильно настроенный, нередко используют как средство рассылки почтового спама.

Взломанные машины. Стандартное ПО. Модифицированное ПО. Смена портов и времени прослушивания. Троянские кони.

Исчерпав возможности по поиску нерадивых администраторов, спамеры примерно год назад или чуть больше начали взламывать любые доступные компьютеры и устанавливать на них одну из вышеперечисленных сервисных служб: SMTP-релей или прокси. Добавьте к этому взрывной рост кабельных подключений в США и какой-нибудь Бразилии (Россия по сравнению с США и той же Бразилией – мелочь), при том что Windows не имеет включенного по умолчанию брандмауэра, администраторы местных кабельных и DSL-сетей никак не защищают своих пользователей в силу низкой квалификации, а сто «сравнительно честных» и хорошо документированных способов взлома незащищенных Windows-машин печатает журнал «Хакер» в каждом втором номере, и вы получите практически безграничное поле деятельности для взломщика. Последняя, и самая мощная, волна взломов исходит от P2P сетей типа Kazaa и e-mail-вирусов, таких как Sobig, несущих в своем коде «рабочий набор спамера».

Надо сказать, что плохая защищенность таких сетей далеко не всегда происходит от низкой квалификации администраторов. Иногда это происходит в силу «политических» причин: вполне квалифицированные администраторы провайдера считают, что они отвечают только за подключение, а все остальное – проблема клиента. Даже в России редко встретишь домашнюю сеть, защищенную брандмауэром и тем более практически невозможно увидеть в памятке клиенту такой сети напоминание о том, что в Windows надо установить брандмауэр.

Однако, установив открытый релей или прокси, спамер рискует тем, что его очень легко обнаружить. Любому администратору достаточно обратиться к подозрительной машине по одному из известных портов и убедиться, что его пускают без авторизации (эта процедура называется «прозвоном»), чтобы внести данную машину в черный список. Поэтому спамеры, особенно в последние полгода-год, начали менять поведение взломанных машин.

Если рассылочный демон принимает обращения только от IP своего хозяина и/или засыпает и просыпается по хитрому алгоритму, и/или постоянно меняет порт, по которому принимаются команды и письма, то прямое обнаружение таких машин методом прозвона обычному администратору сильно затрудняется. Ведь чтобы прозвонить все 65536 TCP-портов потенциально взломанной машины, требуется время – примерно около получаса, за это время она может сменить порт, заснуть и т.д. и т.п.

Однако то, что недоступно постороннему администратору или внешней антиспамерской команде, все еще могут сделать администраторы провайдера. Они могут следить за

странным поведением клиентских машин, которые, прослушав входящее соединение по необычному порту, начинают активно рассылать почту по разным адресам. Такой мониторинг не очень трудно организовать.

Спамерский софт развивается. Относительно невинный софт для прямых рассылок (например, Advanced Mail Sender), в котором спамер в обход сервера провайдера обращается к целевому MTA прямиком с домашнего модема, сменился продвинутыми сложными системами, вершиной которых являются троянские кони широкого спектра действия. Среди их возможностей есть даже апгрейд самих себя, автоматическое распространение, переезд на другие взломанные машины и т.д.

Например, функция такого троянского коня: сходить по HTTP на записанный в нем адрес в заданное время, взять оттуда списки адресов и писем, разослать почту, узнать время и место следующего захода. Иногда троянские программы прослушивают каналы IRC и получают команды оттуда. Это позволяет скрыть источник команд. В отличие от HTTP, где создание сайта или закачка новых файлов отслеживается довольно легко, сообщения на канал IRC могут передаваться через любой из серверов IRC-сети, и для отслеживания источника необходим оперативный доступ к логам всех серверов сразу. В общем, есть много способов скрыть троянскую программу: использовать нестандартные порты, управление, протоколы и т.д. и т.п.

Возможности по активному обнаружению взломанных машин

Теоретически (и практически) способ рассылки, при котором сама взломанная машина обращается по HTTP или IRC за письмами и никогда не находится в режиме прослушивания, труднее всего обнаружить. Практически нельзя понять, что они делают, каков их, так сказать, интерфейс со спамерами,. Например, известно, что некий троян ставит стандартные прокси и SMTP на нестандартных портах. Обычно информация об этом трояне этим и исчерпывается. Зараженных пользователей и их провайдеров интересует только как убрать троян – а антивирусные программы делать это учатся быстро. Для более или менее серьезной борьбы со спамом интереснее знать, кто этот троян распространяет и как он это делает. Для подобных выяснений и полезны администраторы сетей, в которых есть зараженные машины. Например, если троян ходит куда-то зачем-то по HTTP, то во-первых, надо засечь это обращение и его содержание, а также ответ той стороны, а во-вторых, отследить входящие соединения с ним, их источники и суть.

К счастью, у спамеров тоже существует разделение труда – категория «взломщиков» выделилась в отдельную профессию, а товаром и предметом купли-продажи служат списки IP-адресов. Покупателями являются «рассылочники». При этом стандартность установленного ПО играет большую роль. «Рассылочникам» намного удобней работать с использованием обычного списка, не заботясь об особенностях поведения того или иного хитрого трояна. Поэтому сложные и продвинутые троянские кони пока не получили слишком широкого распространения. Впрочем, количество спама пока растет по экспоненте – так что может статься, что уже и получили, только мы этого еще не осознали.

Не исключено, что в конце концов прокси и релеи выйдут из моды, прозвон станет все менее и менее эффективным средством, и единственным способом выявлять очередную черную дыру будет обнаружение спама, посланного из нее.

Организационные усилия по борьбе со спамом

Часто можно услышать о некоем будущем протоколе электронной почты, после внедрения которого спама не станет. Хотелось бы добавить в эту идею немного здорового скепсиса.

Сетевое сообщество не смогло до сих пор внедрить простейшие антиспамерские приемы, которые само же установило в качестве стандарта. Например, разделение портов SMTP-сервера на порт для MTA (25: прием почты от чужого сервера для сохранения своему пользователю; «общение между серверами») и MSA (587: прием письма от своего пользователя для отправки на чужой сервер; «общение между пользователем и сервером»). Эта идея, также как и SMTP-авторизация, появилась именно как реакция на появление спама.

Прошло уже немало времени, однако 587 порт так и не появился в популярных почтовых программах типа Outlook Express или The Bat! А ведь эта простейшая мера позволила бы провайдерам просто закрыть все исходящие соединения по 25 порту и полностью ликвидировала бы прямой спам «по карточкам» – спам с dial-up соединения. Как известно, интернет-карточка стоит 5 долларов, ее хватает на 10 ночных часов, за это время можно разослать десять тысяч писем и спокойно пойти покупать новую карточку, а старую (уже ненужную) заблокирует выведенный из себя провайдер.

Нет никаких технических препятствий так настроить почтовый сервер, чтобы он не принимал почту от «опасных незнакомцев» и блокировал как «спам по карточке», так и черные дыры. Достаточно, например, включить и настроить встроенный в любой SMTP-сервер протокол SSL, так чтобы он отклонял несертифицированные соединения. Сертификаты, идентифицирующие сервер, тоже давно существуют. За 50-100 долларов в год на почтовый сервер можно приобрести их в Thawte или Verisign. К сожалению, при такой настройке вы вообще перестанете получать почту, так как ни у кого, конечно, сертификатов нет.

Новый протокол придумать, наверное, можно. Но работать он будет только в том случае, если на него одновременно перейдут все почтовые системы. Иначе те, кто на него перейдет, окажутся изолированными от тех, кто не перешел. Иными словами, чтобы вы научились плавать в бассейне, кто-то должен сначала налить туда воду. Однако современный интернет напоминает тот самый сумасшедший дом, в котором воду наливают, только после того как вы научитесь плавать.

Вывод: очевидно спам нельзя победить «хорошим» протоколом. Но спам можно побеждать совместными усилиями антиспамерского ПО, систем обратной связи, согласованных действий провайдеров и т.д. И об этом пойдет речь ниже.

Часть 2. Методы борьбы со спамом

Можно встретить разные описания (по сути классификации) средств борьбы со спамом. Поскольку программа это всегда «Алгоритм + Структура Данных», то и классификацию программ правильно основывать на видах используемых данных и используемых алгоритмах. Что мы и попытаемся проделать ниже.

Встречаются, однако, описания, основанные на желании продвинуть свою собственную технологию. При этом часто возникает искаженная картина, вводящая пользователей в заблуждение. Критике таких картин мы также постараемся уделить внимание.

Задача спам-фильтрации

Задача, которую решает детектор спама по содержанию: разделить входящий поток сообщений на спам и нормальную почту, Spam и Ham в английском жаргоне.

Исходные данные

Данные, которые используются для анализа, – это все признаки пришедшего письма. Их можно разделить на четыре пространства, вычисление решений в которых можно производить независимо:

- IP-адрес сервера отправителя;
- оформление и стиль писем, заголовки, форматирование, характерные обороты;
- статистика слов в письмах;
- контрольные суммы («сигнатуры») текстов писем.

Естественно, что пространство признаков по каждому набору данных ограничивают только «интересными» признаками.

Вид данных	**Типичное число признаков, обнаруженных в одном письме**	**Полное пространство признаков**
Оформление и стиль	≈7	до тысячи
IP-адреса «черных дыр»	≈1	сотни тысяч
Статистика слов	≈30	сотни тысяч
Контрольные суммы	≈1	миллионы

Конкретный антиспамовый модуль может использовать все эти пространства признаков или только 1-2 из них. Недостатки и преимущества каждого из пространств признаков мы обсудим ниже. Пока же обратим внимание на необходимое присутствие еще двух составляющих «задачи машинного обучения», классическим примером каковой является детектор спама, а именно: обучающей выборки и обратной связи.

Заметим, что в отличие от пространств слов или элементов оформления, при опознании спама по IP-адресу решение принимается по одному-единственному признаку. Взвешивания по адресу обычно не производится, следовательно, настройка взвешивающего механизма на обучающей выборке не нужна. Однако без обратной связи (в случае с IP – без постоянно пополняемого списка черных дыр) удовлетворительно работающий механизм нельзя построить ни по одному из вышеперечисленных пространств.

Ошибки первого и второго рода

Чтобы любое машинное обучение работало, ему необходимо сообщать об ошибках. Ошибки бывают двух видов. Ошибка первого рода: пропуск спама, то есть пропуск спамового письма. Иными словами – недостаточная полнота метода. Ошибка второго рода – ложные срабатывания, когда не-спам ошибочно относят к спаму. Иными словами – точность метода.

Естественно, приоритет при настройке алгоритма отдается минимизации числа ложных срабатываний. Обычное требование для спам-детектора – уложиться в несколько промилле. Считается, что лучше дать пользователю прочитать несколько спамовых писем, чем скрыть от него настоящее письмо.

Интегральный показатель качества

Процент детектированного спама есть мера полноты, процент ложных срабатываний – мера неточности. Несложно предложить интегральную оценку качества, назовем ее качеством фильтрации. Очевидно, что при точности, близкой к 100%, качество будет примерно равно полноте. Именно полноту фильтрации часто и называют, когда озвучивают те или иные цифры, подразумевая, что точность практически абсолютна.

Надо при этом понимать, что острота восприятия ошибки второго рода зависит от характера поступающих в почтовый ящик писем и индивидуальных предпочтений пользователя: люди, обсуждающие в почте многомиллионные сделки, реагируют на ошибки второго рода гораздо более болезненно, чем сервис поддержки пользователей и, тем более, читатели рассылки анекдотов.

Ложные срабатывания. Разные подходы

Довольно большое значение имеет то, что происходит при ошибках второго рода – от этого зависит величина ущерба, наносимого этими ошибками, и, следовательно, требования к их количеству.

Возможны следующие реакции фильтра на обнаруженный спам:

1. письмо отвергается почтовым сервером; при этом, если оно на самом деле было «законным» письмом, отправитель получит сообщение об этом;
2. письмо помещается в специальную папку; пользователь имеет шанс заглянуть в эту папку и увидеть там ошибочно отфильтрованное письмо;
3. письмо «удаляется», как будто его и не было; никто ни о чем не знает.

Сценарий (3) – самый опасный; к счастью, администраторы почтовых серверов его почти никогда не используют. Однако из популярных текстов, о которых мы будем говорить ниже, зачастую создается впечатление, что используется именно он.

Сценарий (2) с одной стороны имеет тенденцию вырождаться в (3), если качество фильтра хорошее. С другой стороны, регулярный просмотр пользователем папки со спамом снижает пользу фильтрации, хотя это и делается существенно реже, поверхностным просмотром и т.д. В таком сценарии, однако, ущерб от ошибок второго рода минимален, а обратная связь максимальна.

Сценарий (1) – традиционный вариант для «классической» фильтрации по IP адресам. В отличие от (2) он не вырождается в (3), однако при этом нагрузка на сервер существенно возрастает, если в фильтре используется содержимое письма.

Промежуточная зона – «полуспам»

Очень важная, часто недопонимаемая проблема состоит в том, что спам и не-спам пересекаются в очень большой степени.

Рассылки, от которых трудно отписаться, но на которые вы тем не менее (кажется?) подписывались. Подписки, возникающие при регистрации, без вашего ведома. Многочисленные квитанции глупых антиспамерских и антивирусных программ. Автоответчики. Рассылки, совершаемые спамерами при помощи веб-форм из публичных, совершенно неспамерских веб-сервисов, тем не менее слабо защищенных от вторжения. Например, открытки или приглашения вступить в то или иное веб-сообщество – по тексту такого письма даже автор не может понять, спам это или нет. Вся такая корреспонденция может быть смело отнесена к «полуспаму».

Объем этой зоны очень и очень значительный.

Перед началом очередного этапа работ по антиспамовой фильтрации Яндекс провел исследование. Был проведен ручной анализ достаточно репрезентативной выборки из 5151 писем, пришедших на 300 адресов. Так вот, ситуации, когда проверяющий посторонний человек, используя для принятия решения всю мощь своего естественного интеллекта, отнес письмо к такой «промежуточной зоне» составляли до 40 процентов! При этом правило для такого отнесения было достаточно осторожным:

«Полуспамовое» письмо – это письмо от известного проверяющему реально работающего магазина или онлайн-сервиса, в котором пользователь скорее всего регистрировался.

Какой из этого можно сделать вывод? Даже с учетом статистических смещений, характерных для публичной веб-почты, можно попытаться предсказать максимальный теоретический предел качества неперсонализированной спамовой фильтрации. Ведь задача неперсонализированной программы – моделировать поведение максимально объективного незнакомого наблюдателя, не знающего ни про ваши пристрастия, ни про ваши подписки!

Второй вывод таков. Старайтесь не верить заявлениям создателей неперсонализированных антиспамовых продуктов, уверяющих что качество их фильтрации 95 или 98 процентов. В неперсонализированной антиспам-системе, которой известны предпочтения только усредненного пользователя, этот показатель, по-видимому, теоретически недостижим.

Обратная связь

В любом случае ключевой вопрос любой полноценной антиспам-системы состоит в решении, откуда брать сведения об ошибках первого и второго рода. Очевидно, что жалоба на спам или просьба о блокировке адреса – это обратная связь по ошибкам первого рода. Возможна, и крайне желательна, обратная связь и по ошибкам второго рода.

Реализация обратной связи

В интерфейсе большинства современных публичных веб-почт (Hotmail, Yandex, Yahoo, Oddpost) есть специальная папка, служащая для накопления «полуспама» и не очень достоверно определяемого спама, а также кнопка для «реабилитации», сообщающая системе о ложном срабатывании.

В настольных почтовых клиентах, созданных в последнее время, тоже обязательно присутствует обратная связь как первого, так и второго рода. Обычно в виде кнопки «это спам» / «это не спам».

К сожалению, несколько популярных клиентских почтовых программ все еще не поддерживают полноценную обратную связь. Например, все почтовые программы «Микрософт», чей интерфейс и набор возможностей не менялся последние 5 лет, (впрочем, для них написаны многочисленные плагины, способные, пусть и неудобным способом, но восполнить этот недостаток), или некоторые публичные почтовые службы, в которых не реализована обратная связь с пользователем.

Технические приемы на уровне протокола

Особняком от методов, анализирующих только данные пришедшего письма, стоят некоторые довольно популярные в последнее время приемы, задающие особый способ взаимодействия почтовых программ.

1. Незнакомым отправителям посылается письмо типа «Извините, мы с Вами не переписывались, подтвердите, пожалуйста, что Вы не спамер». По приходу подтверждения программа добавляет адрес отправителя в белый список. Есть и известные реализации этой довольно старой идеи: TMDA и WinAntiSPAM.
2. Довольно свежая идея – graylisting («серые» списки). Суть ее состоит в том, что на некоторые письма сервер отвечает не «OK» или «rejected», как обычно, а «временная ошибка». Это само по себе работает (пока) очень хорошо, потому что «хорошие» почтовые серверы через некоторое время повторяют попытку доставить письмо (они обязаны это делать), а рассыльщики спама (пока) этого не делают. Причем можно надеяться, что если спамеры будут пытаться повторять попытки доставки как нормальные серверы, то за это время они успеют попасть в черные списки. Время повторного соединения обычно полчаса, и это, в общем, некритично, тем более что оно относится только к первой корреспонденции между двумя незнакомыми сторонами, так как ранее проверенные адреса не проверяются, а запросы на проверку кэшируются и вновь не посылаются.
3. Проверка корректности адреса отправителя (envelope-from). Проверку существования домена в большинство серверов вставили очень давно, и до сих пор она изредка срабатывает, хотя эффективность ее ныне невелика. Сейчас многие стали вставлять проверку адреса целиком. Хотя это довольно накладно по ресурсам – для этого надо связываться с сервером, на котором расположен адрес, и осмысленный ответ при этом не гарантирован, – однако, по крайней мере пока, это неплохо работает.

Алгоритмы

Как видно из приведенной таблицы, потоки данных сильно отличаются для разных типа признаков. Рассмотрим их по отдельности

Проверка IP. DNS-зона. Имя черного списка как интегральный признак

Простейшая в реализации, и, безусловно, именно поэтому самая популярная – фильтрация по пространству IP-адресов. Для каждого письма проверить надо 1 (редко больше) IP-адрес, делается это сейчас при помощи специальной DNS-зоны для каждого из черных списков. Поиск в DNS, в сущности, – простая хеш-функция. Часть из списков разрешено скачивать, и для эффективности такие зоны разумно создать на локальном DNS-сервере.

Что еще характерно для данного пространства признаков? Во-первых, отлично отработанная обратная связь.

Во-вторых, это самое нестабильное и текучее пространство признаков, для которого характерно постоянное исчезновение и добавление адресов. Следовательно, считать индивидуальный весовой коэффициент для каждого IP довольно дорого и не очень эффективно: данных слишком мало, а адреса все время меняются.

Отсюда и простейший способ понижения размерности этого пространства – заменить индивидуальный IP-адрес на список, в котором он обнаружен. Принципы формирования, надежность и применимость списков в первом приближении можно считать униформным для всех «его» IP-адресов.

Низкая стоимость вычислений, простота и налаженность процедуры обмена данными и их небольшой объем, однозначность данных (IP практически невозможно подделать). Все эти факторы играют решающую роль в доминировании данного признака в антиспамовом ПО.

Байесовская фильтрация по словам

Очень простым, интуитивно понятным методом «машинного обучения с учителем» (то есть при наличии Spam&Ham выборки) является наивная байесовская классификация. «Наивной» она называется потому, что исходит из предположения о взаимной независимости признаков, и, как ни странно, этого часто оказывается вполне достаточно. Использование формулы Байеса для фильтрации спама [http://www.paulgraham.com/spam.html] предложено совсем недавно, примерно год назад.

Автор, Paul Graham, предназначал его для персональной фильтрации. Для работы требуется, чтобы у классифицируемого объекта было достаточно признаков. Этому требованию идеально удовлетворяют все слова (или токены) писем данного пользователя, исключая разве что очень редко встречающихся и совсем короткие. Вторым требованием является постоянное переобучение и пополнение коллекции Spam+Ham. Все такие условия идеально работают в локальных почтовых клиентах, поддерживающих этот алгоритм.

К сожалению, использовать метод Байеса прямо, непосредственно в условиях массовой почтовой службы, затруднительно, в основном по причине большого разнообразия словарного состава клиентских ящиков. Так, из-за того, что в обучающей выборке наверняка будет очень много туристического спама, все письма, например, из турагентства могут быть отнесены к спаму. Не смогут здесь помочь и другие методы классификации текстов по словам, более традиционные для науки информационного поиска (например метод Роккио или метод опорных векторов). Однако как-то использовать вероятность отнесения письма к среднестатистическому спаму (или иную меру текстуальной схожести), полученную анализом словарного состава, по-видимому, можно и в массовых сервисах.

Генетические алгоритмы и ручное выставление весов

В результате больших усилий многих людей было выявлено огромное количество различных эвристик, связанных с особенностями заголовков спамерских писем, их оформления, характерных стилистических оборотов, типичных фраз. Суммарное количество подобных признаков у известного фильтра SpamAssassin, например, приближается к тысяче. К сожалению, несмотря на то, что практически каждое спамовое письмо содержит хотя бы несколько таких признаков, над пространством таких признаков невозможно построить устойчивый Байесовский автомат. Причин здесь две:

1. слишком мало число признаков, типично встречающихся в одном письме;
2. отсутствует балансировка, то есть нет достаточного количества признаков не-спама.

В этих условиях применяют другие алгоритмы. Например, SpamAssassin применяет генетический алгоритм. В нем подбор начинают со случайной простановки весов для каждого признака (создание «хромосом»), а затем «скрещивают» и «мутируют» хромосомы в поисках оптимальных значений весов для данной тестовой выборки. Оптимум (в теории) может оказаться не глобальным, а локальным, но этого обычно более чем достаточно.

Часто практикуется и ручное выставление весов для каждого признака, ведь количество их обозримо, и опытные администраторы в состоянии контролировать и постоянно корректировать спам-фильтрацию для почты своей компании.

Обнаружение повторов и признак массовости

Если антиспамовая система имеет дело с большим потоком писем, она может и должна пытаться находить повторы писем. Во-первых, так можно вылавливать письма, уже известные (помеченные ранее) как спам. Во-вторых, массовость письма сама по себе является неотъемлемым признаком спама. Из утверждения, что письмо есть спам, неизбежно следует, что оно массовое. Таким образом, признак массовости есть необходимое, хотя и не достаточное условие спама.

Строго говоря, одиночные нежелательные письма тоже можно считать спамом, но бороться с ними имеет смысл одиночными же методами, поэтому для данной статьи можно смело принять такое допущение.

Интересной темой является практическая реализация выявления массовой корреспонденции. Попытки наладить распределенные системы обмена контрольными суммами писем, предпринимаемые в рамках таких проектов, как DCC (несколько контрольных сумм по тексту и заголовкам письма) или Бритва Вайпула (одна «нечеткая» контрольная сумма) в настоящий момент упираются в общие ограничения P2P-технологий по производительности. Дело в том, что для того чтобы обеспечить статистику повторов в реальном времени, участники системы вынуждены поддерживать режим постоянного обмена этой информацией. В момент спамовой атаки скорость реакции таких систем становится неприемлемо низкой. Видимо, об эффективном применении систему детектирования повторов можно пока говорить только в системах с очень большим потоком писем, у крупных провайдеров или на публичных почтовых серверах, например веб-почты.

Различным методам выявления повторов будет посвящена последняя часть этого сообщения. Пока можно лишь заметить, что признак массовости служит неплохим фактором и сам по себе, и в различных интегрирующих системах.

Интегрирующие системы

Ни один отдельно взятый набор признаков не в состоянии обеспечить максимальное качество фильтрации. Очевидно, преимущество здесь окажется у систем, интегрирующих решения по всем пространствам признаков.

Пионером здесь является SpamAssassin, который позволяет применить как генетический алгоритм, так и ручное взвешивание поверх не только собственного или «настроенного» набора флагов, но и с учетом байесовского текстового подобия, и с учетом взаимодействия с DCC-модулем детектирования рассылок.

Отдельным вопросом является то, какой алгоритм должен работать в точке окончательного принятия решения.

Точки применения фильтра

Кроме различия в исходных данных, алгоритмах и видах обратной связи, антиспамовые средства надо различать по месту их применения. Таких мест можно выделить два: почтовый сервер и клиентский компьютер.

Фильтрация на сервере: царство IP-метода

Сервер характеризует большим поток писем, на нем можно обеспечить гарантированную производительность, на нем есть постоянная связь с другими серверами. При превышении потоком писем некоторого уровня можно начать детектировать рассылки. На серверах, по-видимому, неприменим в чистом виде байесовский алгоритм по тексту письма (см. выше).

Однако наиболее стандартным, легко реализуемым и относительно эффективным методом является фильтрация по IP, и с учетом этих обстоятельств этот метод в настоящий момент доминирует. Можно ожидать появление средств фильтрации и по другим признакам.

Препятствием внедрению методов, основанных на анализе письма, служит дилемма диагностирования и обратной связи. Предположим, что на сервере не поддерживаются пользовательские папки для накопления спама. В этом случае сервер обязан выдавать диагностику (550) на все без исключения отфильтрованные сообщения по мере их получения, что накладывает к системе анализа жесткие требования по производительности.

Фильтрация на клиенте: царство Байеса

У клиента совершенно другая картина. Здесь малый поток данных, неизвестная производительность компьютера, отсутствие постоянной связи с интернетом – то есть невозможно или слишком дорого постоянно «закачивать» массивы контрольных суммы писем или IP черных дыр. Зато очень точно можно отличить чужие письма, они всегда не похожи на ваши просто по тексту; «вкусы» одного пользователя выяснить легко. По всем этим причинам клиентские антиспамовые программы представляют из себя «царство Байеса».

Часть 3. Осторожно, маркетинг

Как мы уже говорили, при описании и классификации средств борьбы со спамом следует исходить из различий в используемых данных, алгоритмах и способах обратной связи. Однако зачастую можно встретить некорректные описания антиспамовых средств, их возможностей и спектра применения, вызванные маркетинговыми причинами. В частности, создатели различных программных продуктов публикуют статьи, в которых практически всем методам фильтрации кроме своих, приписываются несуществующие

недостатки и ограничения. Нам бы хотелось в этом разделе защитить реноме этих методов.

Антиспамовые методы на стороне провайдеров

1. Из рекламных статей прежде всего нельзя понять, что происходит при отфильтровывании письма по IP-адресу. Читателям, по сути, внушается апокалиптическая картина, что письма проваливаются в никуда; многомиллионные контракты срываются и т.д. и т.п.

   Однако провайдеров, которые ведут себя по таком сценарию (сценарий (3) – см. выше), на практике не существует (мы не знаем НИ ОДНОГО такого провайдера). Все известные нам почтовые серверы отвечают внятной диагностикой (возвращаемой отсылающим сервером автору письма) на попытку соединиться с IP-адреса из черного списка. Например (в худшем случае):

   Your message to cmail.yandex.ru was rejected.I said: RCPT To:iseg@yandex-team.ru
   And cmail.yandex.ru responded with 550 5.7.1 iseg@yandex-team.ru… Spam source.

   Если же список черных дыр официально публикуется и поддерживается, то в диагностическом сообщении SMTP-сервера принято указывать еще и URL страницы, на которой можно получить подробное объяснение, почему данный IP-адрес попал в черный список. Более того, все известные нам скандалы в Рунете, связанные с блокированием, возникали именно тогда, когда «официальные спамеры» получали такую квитанцию и начинали чувствовать себя ущемленными.

2. В продолжение этой же идеи провайдерам приписывается использование неких тайных, нигде не публикуемых, секретных черных списков.

   На самом деле никаких «тайных» списков, конечно же, нет.

   У провайдеров действительно есть свои собственные списки, которые они не публикуют, главным образом потому что публикация – это дорогостоящий шаг, требующий регулярного обновления, поддержки, в общем, некоторого ресурса. Кроме того, в нашей стране не очень принято публиковать нелицеприятные заявления о каких-то компаниях (чем, по сути, является публичный «черный список»). На это надо отдельно решиться.

   Кое-что делалось в этом направлении – была такая инициатива DRBL [http://www.agk.nnov.ru/drbl/]. Однако то, что получалось, видимо, было слишком сырым, чтобы это использовать массово. Тем не менее любому пользователю, отправившему письмо с заблокированного адреса, придет серверная квитанция в случае недоставки, с четким указанием причины отказа в сервисе – «ошибка 550, отказ в соединении, источник спама» – см. выше. Правда, это сообщение обязано быть на английском языке.

   Таким образом, эти данные никоим образом не скрываются. Такое поведение требуется по стандарту протокола SMTP.

3. Отсюда же проистекают утверждения, что провайдеры постоянно ведут между собой войны, что из-за этого возрастает число писем, ошибочно принятых за спам, так как в черные списки часто попадают провайдеры в целом, что клиенты

воюющих сторон лишаются возможности общения друг с другом. В сущности, публичные IP-списки объявляются принципиально ненадежным методом фильтрации.

Это не совсем так или даже далеко не так.

Действительно, некоторые списки составляются не только для фильтрации как таковой. Иногда в этой деятельности присутствуют элементы «борьбы со спамом». Представим себе дворника, в задачу которого входит содержать определенную территорию в чистоте; он может активно подметать, а может гоняться за мусорящими гражданами и больно бить их по голове (а также заниматься и тем, и другим вместе). Такое битье само по себе чистоты не добавляет, но может (как некоторые думают) благотворно сказаться на ситуации в будущем. Списки, в которых встречаются «дружественные к спаму провайдеры», в большой степени представляют собой именно средство битья по голове, и об этом все прекрасно знают.

Например, известный список SPEWS постепенно расширяет заблокированную область с конкретного виноватого IP до его сети, потом до вышестоящей сети, иногда доходя до блокирования всех сетей данного провайдера.

Однако есть очень много списков, где никаких списков адресов «дружелюбных к спаму провайдеров» нет, и эти списки всем известны и отлично пригодны именно для фильтрации спама как такового.

4. Про сервис IP-фильтрации SpamCop, можно услышать, что это не фильтрация спама, а всего лишь средство социального типа, позволяющее «наехать» на провайдера.

   На самом деле SpamCop – эффективное средство оперативного обнаружения и блокирования источников рассылки спама, позволяющее отсекать эти источники прямо в процессе рассылки. Он не только фильтрует почту для своих клиентов, но и публикует очень неплохой черный список «спамерских» IP. Для определения процента спама с того или иного сервера SpamCop использует статистику трафика от доверенных почтовых серверов, по-видимому, преимущественно расположенных в США, поэтому доля спама для русского трафика у него несколько переоценена. Тем не менее применять SpamCop для фильтрации почты, приходящей с зарубежных машин (а это большая часть русского спама), можно с успехом.

5. Можно встретить утверждения, что применение у провайдеров единственного вида фильтрации, фильтрации по IP, обусловлено тем, что проверять содержание писем провайдеры не могут по закону. Отсюда делается вывод, что фильтрацию правильней организовывать у третьей стороны, например, у владельцев того или иного программного продукта.

   Причина, конечно, не в законе.

   Главная и основная причина чисто техническая – гораздо проще проверить IP (4 байта) по списку, чем заниматься анализом содержания. Кроме того, IP-адрес отправителя – это единственный параметр письма, в выборе которого спамер ограничен. Любой провайдер, при наличии соответствующих программных средств

и/или при получении заказа на такую услугу, вправе и в состоянии (если посчитает это экономически оправданным) организовать техническую фильтрацию по любому полю письма.

Что касается подразумеваемой проблемы перлюстрации и якобы существующего запрета со стороны Закона о Связи даже на технические проверки содержимого, то рассуждения эти являются чисто спекулятивными, так как перлюстрация подразумевает контроль содержимого человеком или для человека. Никаких судебных прецедентов на эту тему до сих пор не было. Несомненно, этот усиленно муссируемый вопрос с теми же основаниями можно отнести и к проверке письма на вирусы, что показывает его абсурдность

Непонятно также и каким образом программный продукт, пусть даже и внешний по отношению к провайдеру, может помочь ему обойти Закон о Связи, в котором явно прописан запрет на переделегирование полномочий.

6. Встречаются в статьях и заведомо заниженные показатели эффективности IP-фильтрации, например про черные списки можно услышать, что они фильтруют не более 25-30% спама.

   Совершенно непонятно, откуда берутся эти оценки. Фактически IP-фильтрация фильтрует больше, чем 25-30% спама. В докладах первой международной конференции по борьбе со спамом [http://spamconference.org/] в январе 2003 утверждалось, что черные списки фильтруют до 60 процентов спама, по опыту Яндекса [www.yandex.ru/monitoring]этот показатель составляет около 35-40%.

О методах фильтрации спама в корпоративной сети

Можно услышать, что на корпоративном почтовом сервере, установленном в офисе или у провайдера, неприменимы «провайдерские» средства фильтрации по IP.

На самом деле для корпоративной почты нет ничего невозможного в отсеивании писем по тем же самым черным спискам IP-адресов. Единственная разница – в случае получения всей входящей почты на корпоративный почтовый сервер через сервер провайдера, некоторый спам придется сначала принять (необходимо прочитать IP-адреса из заголовков, так как входящий IP фиксирован), и лишь потом, после получения первых нескольких заголовков, ответить все тем же самым Reject-ом. Многие компании с успехом применяют именно такие «провайдерские» методы фильтрации спама.

О методах фильтрации спама на клиенте

Утверждается, что хотя локальные антиспамовые программы работают по отзывам прессы довольно хорошо, с русским языком они не работают, так как ориентированы только на спам американского происхождения.

Это неверно. Действительно, в словаре таких программ, входящем в поставку, может не быть или быть мало русского спама. Однако их главная сила именно в обучаемости. Достаточно в течение 1-2 недель регулярно пользоваться кнопками «спам»/«не-спам» и процент отсекания спама резко вырастает. В этом отношении и на английской почте байесовские программы работают ничуть не лучше, хорошие результаты возникают только после постоянного и аккуратного «доучивания» системы.

Mozilla Mail (простейший Байесовский алгоритм), например, превосходно работает с русским языком и начинает отсекать более 90 процентов спама без ложных срабатываний очень быстро.

О методах фильтрации спама в крупных веб-почтах

Можно услышать, что серверы публичной почты относятся по типу используемых антиспамовых средств к категории «фильтрации на клиенте». Между тем, по всем признакам – это полноценный серверный механизм, которому доступны в бесперебойном режиме все виды данных и все виды обратной связи.

Утверждается также, что на таких серверах существует техническое средство, позволяющее заметить факт массовой рассылки на адреса, зарегистрированные в системе. На самом же деле нет никакой технической возможности, простой или характерной именно для веб-почт, заметить факт массовой рассылки. Дело в том, что спамерские серверы, особенно в последние годы, практически никогда не используют отправку большого количества писем в одной SMTP-сессии. Эти времена давно прошли. Так поступают для эффективности только настоящие серверы легальных рассылок, например Subscribe.ru. Спамеры же используют множество соединений и, более того, множество релеев (см часть 1), для отправки одного и того же письма по многим адресам.

Что же есть на самом деле? Есть все тот же упомянутый выше механизм детектирования дубликатов, доступный **любому** почтовому серверу с большим трафиком, будь то провайдер или крупная корпорация. Надо принять письмо, проанализировать его содержание, сравнить с уже присутствующими в базе письмами, используя при этом самые обычные методы обнаружения массовых рассылок: DCC, Бритву Вайпула или «Супершингл Яндекса» – подробнее см. последнюю часть данной статьи.

О лингвистическом методе, работающем на опережение

Зачастую в маркетинговых публикациях проходит сквозной нитью мысль о неэффективности обратной связи, например, встречаются утверждения что технология, базирующаяся только на образцах и голосованиях пользователей по уже разосланным письмам, всегда слегка опаздывает. Ей противопоставляется Идеальная Лингвистическая Технология, Которая Никогда Не Опаздывает.

Отсюда и необъяснимая классификация методов фильтрации на «Формальные методы» и загадочные «Лингвистические методы» и обещания, что продукты, построенные на Идеальной Лингвистической Технологии, Которая Никогда Не Опаздывает, будут фильтровать до 90-95% спама и утверждения, что они работают по схеме «черного ящика», фильтруют «под ключ» и т.д.. Вопрос: как же такие системы собираются получать обратную связь?

Маленькое заключение

В свете всего, что было сказано выше про спам и технические средства его обнаружения, представляется, что не существует таинственных методов, дающих небывалый процент улавливания спама. Более того «Идеальный продукт» обязан использовать данные обратной связи и постоянно обучаться. Любая жалоба пользователя, любая просьба о реабилитации письма имеют отношение ко всему комплексу анализируемых факторов (IP, слова, флаги, контрольные суммы) и должны мгновенно и по возможности автоматически обрабатываться.

Часть 4. Детектирование массовых рассылок на Яндекс.Почте

О почтовой службе Яндекс.Почта

На Почте Яндекса письма проходят три уровня фильтрации.

На первом этапе отбрасывается явный спам – сообщения, приходящие от неадминистрируемых (взломанных, открытых) почтовых серверов, либо пойманные в спамовые ловушки.

Затем каждое письмо проверяется антивирусной программой DrWeb. При этом зараженные письма, не содержащие ничего, кроме самого вируса, уничтожаются, а зараженные письма с текстом помечаются «Осторожно, вирус!».

Последним работает фильтр, помещающий в папку «Рассылки» подозрительно похожие письма, разосланные по слишком большому списку адресов.

На странице http://mail.yandex.ru/monitoring/ публикуются ежедневные данные, по которым можно следить за ходом борьбы со спамом на Яндекс.Почте.

Обратная связь

На Яндекс.Почте реализованы (благодаря наличию специальной папки «Рассылки») оба вида обратной связи, как по ошибкам первого рода (Кнопка «ФУ! ЭТО СПАМ»), так и по ошибками второго рода: ссылка «Это не рассылка» в папке Рассылки.

Зачем детектировать повторы?

Многочисленные повторы текста некоторого письма сами по себе не есть спам. Это могут быть технические рассылки самой разной природы, например, счета за мобильный телефон или письма, уведомляющие о важной регистрации. Однако, как писалось выше, спама не бывает без повторов, т.е. массовость – важный родовой признак спама. Заметим, что определение повторов важно не столько и не только как отсекатель известного «заведомого» спама (надежно детектированного иным методом: например черным IP, ловушкой spam-trap), но и в процессе принятия решения и вообще при любой классификации корреспонденции. В частности, на Яндекс.Почте этот признак в настоящий момент (сентябрь 2003) используется для направления корреспонденции в папку «Рассылки».

Что такое контрольная сумма? fnv, md5, crc

Контрольная сумма (или «сигнатура») – это уникальное число, поставленное в соответствие некоторому тексту и/или функция его вычисления. Функция вычисления контрольных сумм может преследовать несколько целей: например «невзламываемость» (минимизируется вероятность того, что по значению контрольной суммы можно подобрать исходный текст) или «неповторяемость» (минимизируется вероятность того, что два разных текста могут иметь одну контрольную сумму). Существует обширная литература по алгоритмам вычисления контрольных сумм, я упомяну здесь самые известные: fnv, md5, crc. Обычно более-менее все равно, какой из них выбрать, но в любом случае при выборе алгоритма его положительной стороной можно считать хорошее быстродействие.

Нечеткие дубликаты. Постановка задачи

Однако даже при наличии быстрой, не взламываемой и точной функции проблему нельзя считать решенной. Дело в том, что повторяющиеся письма очень часто незначительно отличаются, в результате для двух писем, разнящихся, предположим, на одно слово, получатся две совершенно разные контрольные суммы. Не вдаваясь в ситуацию активного противодействия спамеров системам обнаружения спама (этому чуть ниже будет посвящен отдельный пункт, содержащий небольшой анализ), отметим, что наиболее типичная ситуация для порождения разных писем в любых рассылках – вставка имени получателя в текст и заголовок.

Опыт современных поисковых систем

Задача, схожая с этой, но на гораздо больших масштабах данных, уже встречалась в нашей компании, когда нам приходилось решать проблему «почти дубликатов» в веб-поиске. И хотя тот алгоритм (представленный на всемирной конференции по интернет-вычислениям WWW2002 на Гавайях) не годился в использовании напрямую, однако общий круг идей и методов был нам хорошо знаком.

Шинглы

Наиболее известным способом обработки почти-дубликатов в веб-поиске, изящно изложенным Андреем Бродером в 1997 году, является метод «шинглов». Очевидно, чтобы повысить вероятность того, чтобы в результате небольших изменения текста контрольная сумма не изменилась, можно попытаться выбрать из текста несколько подстрок. Шингл (от английского shingle – чешуйка, черепичка) – это и есть подстрока текста, по которой происходит вычисление контрольной суммы.

Выбирать такие подстроки можно по-разному. Во-первых, можно брать разный шаг, например: символ, слово, предложение. Во-вторых, решить, как они должны идти – внахлест (как раз так и получаются именно «шинглы») или встык. В-третьих, следует понять, какого размера должны быть подстроки: выбранный размер должен свести к минимуму случайные повторы, то есть должен быть достаточно большим. При этом он должен оставаться и достаточно малым, чтобы типичные изменения текста не разрушили большую часть сигнатур. Конкретные цифры я здесь не привожу, по понятным причинам они не должны афишироваться. В-четвертых, надо решить, делать ли их фиксированного размера. И, в-пятых, поскольку возможных подстрочек в тексте чересчур много, надо выбрать – какие запоминать, а какие выбрасывать.

Встык

Если запоминать контрольные суммы для строчек фиксированной длины, идущих встык, то вставка и удаление одного символа (особенно в начале текста) разрушит их все, как их ни выбирай. Это, безусловно, самый неудачный вариант.

Однако, если отменить фиксацию длины и считать подстрочки от одной характерной точки в тексте до другой (например, от буквы «ю» до буквы «ю», или от двухбуквия, сумма численных значений символов (букв) которого кратна 50, до следующего такого же), то вставка (или удаление) с большой вероятностью разрушит только тот шингл, где она случилась.

Когда заведомо известно, что документ изменяется, пусть и сильно, но в малом количестве мест, такой тип сигнатур успешно применяют. Например: передача однотипных HTML-файлов прокси-серверами или синхронизация репозитория исходных текстов программ.

К сожалению, в этом варианте сигнатур остается слишком много, если, конечно, не выбирать характерные точки, отстоящие друг от друга в среднем далеко. Но тогда строчки становятся слишком большого размера, а алгоритм становится слишком неустойчив к небольшим изменениям текста. Для вероятностного сравнения двух документов все равно необходимо сокращать выборку, и об этом позже.

Внахлест

Поначалу кажется, что считать контрольные суммы по всем строчкам внахлест – странная идея. Нам же нужно сократить объем данных для сравнений, а в таком варианте он страшно возрастает? Однако именно так мы гарантируем, что не пропускаем ни одной подстроки текста (заданной длины) и, при условии, что получится придумать устойчивый способ отбирать шинглы, нам удастся очень точно отождествлять документы, имеющие совпадающие части.

Выборка. Какие шинглы запоминать?

Классический алгоритм Бродера предлагает отбирать либо неизменное количество минимальных по значению шинглов, либо все шинглы, значение которых делятся на какое-нибудь небольшое число (10-30). В первом случае мы получаем фиксированную по размеру выборку (что иногда удобно) и приличный по размеру набор шинглов даже для относительно коротких документов, но, например, нельзя будет судить по наборам шинглов о вложенности документов друг в друга. Во втором случае число шинглов пропорционально размеру документа, то есть оно переменное, что неудобно, зато можно по набору шинглов оценивать такие интересные вещи, как вложенность документов друг в друга или процент их пересечения. Наконец, последний, самый «модный» алгоритм формирует фиксированную выборку, размер которой определяется заданным числом (85 для веб-документов) разных независимых случайных функций, для каждой из которых запоминается ровно один шингл, минимальный по значению контрольной суммы. Этот подход комбинирует преимущества двух предыдущих.

Короткие документы. Что можно сделать?

Что делать с совсем короткими документами, для которых алгоритм отбора шинглов (например второй) может вообще не выбрать ни одного подходящего? Или выбрать слишком мало? Мы знаем два альтернативных решения, одно из них: закольцевать текст документа, то есть виртуально продолжить его начало после окончания, чтобы добиться получения необходимого количества шинглов даже в таких условиях. Второй подход, применяемый в Яндекс.Почте, состоит в использовании выборки, размер которой имеет логарифмическую зависимость от размера документа.

Супершингл

Если для каждого письма отбирать более одного шингла, мы столкнемся с задачей отождествления документов, имеющих только несколько совпавших шинглов. Как бы мы ни сокращали число шинглов, все равно остается нетривиальный объем работы: данных

очень много, даже если отбрасывать слишком редкие и слишком частые шинглы; не существует мгновенно работающего запроса по отождествлению документа и т.д.

Поэтому на практике часто над набором шинглов документа считают еще одну контрольную сумму, так называемый «супершингл». Очевидно, тогда совпавшими будут считаться только документы с полностью совпавшими наборами шинглов. Однако при правильном подборе алгоритма и его параметров этого может оказаться достаточно и для работы неплохого детектора рассылок. Задача будет сводиться к вычислению всего одного числа и нахождению его в простейшей базе данных.

Замена супершингла: лексические сигнатуры

Совсем необязательно искать очень похожие документы по контрольным суммам и хитрым подстрочкам. Вполне успешно (по крайней мере в задачах веб-поиска) работают и лексические (основанные на словах) методы. Все разнообразие этих методов сейчас разбивают на два класса: локальные и глобальные лексические сигнатуры.

Если локальные сигнатуры рассматривают документ изолированно от коллекции и пытаются извлечь несколько характерных слов, основываясь только на их статистике в самом документе – TF (характерный пример: взять 5 самых частотных слов в документе длиннее пяти букв и упорядочить их по убыванию частоты), то глобальные либо пытаются при анализе документа учитывать информацию о глобальной статистике слова – IDF, либо вообще выбирают опорные слова, опираясь исключительно на уже существующий инвертированный индекс (см. метод Яндекса на WWW2002). Для работы глобальных методов необходимо как-то считать общую статистику слов, что в интенсивной антиспамовой системе вполне возможно, например, в рамках байесовского подхода.

Антидетекторы. Борьба борьбы с борьбой

Рассмотрим несколько типичных способов, с помощью которых спам-программы могут пытаться обходить детектор рассылки. Речь идет, конечно же, об автоматической генерации небольших изменений для каждого письма или группы писем.

Эту автогенерацию можно разделить на несколько категорий, механизм детектирования которых рассмотрим по отдельности.

1. Генерация невидимого (или очень слабо видимого) текста средствами HTML-форматирования.

   В этом случае детектирование рассылок по контрольным суммам может быть полностью разрушено. Однако чтобы добиться такого эффекта, спам-системам придется интенсивно пользоваться разными приемами HTML. Существует целый букет эвристик, связанных с оформлением письма, надежно распознающий эту технику. Это и отсутствие plain-text части и масса специфичных тегов HTML или нестандартные стилей CSS (например visibility: hidden). В любом случае здесь речь идет не столько о расчете сигнатуры, сколько о хорошем детекторе особенностей html-формата.

2. Генерация видимого «мусора», то есть случайных буквенных цепочек, добавляемых в заголовки и текст письма.

В этом случае существенно помогает исключение из шинглов «несловарных» слов (по сути, приравнивание их к пробелу). Обратите внимание что «словарь» в данном случае – это не канонический словарь русского языка Ожегова, а частотный словарь, накопленный по реальным письмам. Кстати, доля несловарных слов с таким «антидетектором» будет необычно высокой, а это может послужить отдельным неплохим признаком.

3. Вставка пробелов в текст в случайных местах внутри слов и удаление их между словами. Против такого приема может помочь подсчет шинглов с гранулярностью в один символ с предварительно удаленными пробелами (все слова текста склеить в одну цепочку из букв, фиксированным окошком вычислить шинглы). Кроме того, доля «несловарных» слов с таким антидетектором тоже будет аномально высока.
4. Вставка значащих слов в текст в случайных позициях. Этот вид антидетектора редок, так как затрудняет понимание текста письма. Генерировать же бесконечное количество синтаксически связанных перефразирований спамеры еще не научились. В любом случае с таким антидетектором остается надеяться на снижение эффективности спама и, соответственно, существенное повышение цены вхождения в этот рынок.

Низкий порог срабатывания

Даже с учетом того, что супершингл с большой вероятностью склеивает два документа, отличающиеся на одно-два («значащих») слова, даже с учетом всех возможных методов очистки и препроцессинга, показатели эффективности супершингла на Яндекс.Почте (45-60%) кажутся слишком высокими. В чем же дело?

Дело в том, что букет писем с наложенными автосгенерированными изменениями кластеризуется (собирается) пусть и не в один супершингл (это был бы недостижимый идеал), но в относительно небольшое количество супершинглов. С учетом огромного спам-трафика на Яндекс.Почте и аккуратно установленного, достаточно низкого порога срабатывания по числу повторов, очень многие такие кластеры переходят этот порог.

Заключение

Детектор массовых рассылок внедрен в Яндексе в ноябре 2002 года. Мы продолжаем его совершенствовать и считаем, что это относительно простой в реализации, но эффективный механизм, предназначенный как для облегчения ежедневной работы пользователей с почтой, так и для использования его в составе более сложной антиспам-фильтрации.

Не существует рассылок, на которые нет жалоб пользователя. Не существует спама, который люди не просят реабилитировать. Границу часто провести невозможно. Следовательно, даже после открытия пользователю понятного интерфейса по обучению системы («ЭТО ПИСЬМО = СПАМ», «ЭТО ПИСЬМО = НЕ СПАМ») и налаживанию сбора всей информации, следующим шагом должна быть максимальная индивидуализация антиспамовой системы.

И еще. Не стоит путать спам и нежелательную почту. Да, не все в жизни происходит так, как хочется; в частности, кое-кто шлет ерунду, которую и читать-то смысла нет. Это не означает, что эта ерунда – спам. Не надо ждать от антиспамового фильтра ни решения всех жизненных проблем, ни превращения почтового ящика в интереснейшее или захватывающее чтение, от него надо ждать всего лишь исчезновения спама.