

Federal Trade Commission
Title: CAN-SPAM ANPR
Subject Category: CAN-SPAM Act - Advanced Notice of Proposed Rulemaking
("ANPR")
Docket ID: [3084-AA96]
CFR Citation: 16 CFR 316

Further comments of Stephen Satchell, Incline Village, NV 89450-6900

Reference: <https://secure.commentworks.com/submitcomment.aspx> section C

Ladies and Gentlemen of the FTC and the public,

My comments on the 10-day implementation period change would not fit into the space provided in the original form, because I feel that there is significant background that needs to be established to show why such a long implementation period is inappropriate for electronic mail.

My background includes my being a former employee of Addressograph-Multigraph, known at the time of my employment as AM International.

SUMMARY

This paper looks at his guess at the reasoning behind the Congressional dictate of 10 days for updates, the fallacy of such a loose schedule in light of existing Internet technology, and a proposal for a new update deadline requirement.

REMOVAL AND PAPER FULFILLMENT

The author believes that the 10-day update requirement is taken from Congress' understanding (from testimony) that it can actually take that long to remove an address from a mailing list. When one looks at the management of a physical mailing operation, it becomes obvious why it can take that long. Let's look at the process:

The process of fulfillment (preparation, bundling, and delivery to the USPS) of paper-based commercial mail pieces such as flyers, magazines, newspapers, catalogs, and brochures is mature, and the practices are widely known. Depending on the number and variety of mail pieces prepared and shipped, the practices may vary widely from one operation to the next. Many of the processes involved in affixing the mailing address to mail pieces has been the focus of attempts at automation.

In its basic form, a human being writes or types an address on a mailing piece, and places

the mailing piece (perhaps after stuffing) in a basket for subsequent collection. This method is still used in many offices around the country and around the world to deliver commercial messages, both advertising and transactional.

Automation has taken many forms:

- ⑩ The use of window envelopes, so that the address printed (or pre-printed) on a bill, letter, or other mail piece will also provide a mailing address
- ⑩ Preprinted “sticky-labels” that are prepared separately from the mail pieces and applied either by hand or machine to the outside of the mail pieces; these labels could be printed from printing plates, or generated by the means of a computer and suitable printer
- ⑩ Preprinted envelopes, bags, or other containers that are prepared separately from the contents of the mail piece, and “stuffed” before sending it on its way
- ⑩ “Addressing machines” such as the Addressograph(R) to directly print the address on each mail piece, by means of a metal or plastic plate embossed with the address of the recipient
- ⑩ Use of direct-print technology, such as ink-jet printers, to directly print or inscribe the postal address of the recipient on the mail piece

Many of these technologies require the off-line preparation of addressing information: printing labels, printing envelopes or other enclosures, preparing address plates, or preparing paper or magnetic tape. The off-line preparation requirement is a limitation of the equipment currently in use, and the cost of updating the equipment to that which uses on-line transactional sources of addresses may be prohibitive. In the case of certain high-speed equipment, the data speed requirement is higher than a computer can be reasonably expected to meet, and so off-line preparation removes the speed requirement from the computer holding the data-base of information.

Off-line preparation of address information means there is necessarily a delay between the time a removal request is received and the time the address is no longer affixed to a mail piece. Existing supplies of preprinted mailing labels need to be used up. The address plate for the addressee has to be located and removed. The source data for paper or magnetic tape has to be deleted, and the paper/mag tape recreated. The backing store need to be reloaded with corrected data.

Address updates in a paper-piece preparation environment can also severely impact the work flow, so a once-a-week update cycle permits the maximum use of equipment and the highest productivity, and thus lower costs.

In a paper environment, there are several ways that a recipient can request removal:

- ⑩ Call a telephone support operation and request removal
- ⑩ Write a letter and request removal

⑩ Use a Web site to request removal; usually this generates a letter

The required time to delete an address, ten days, includes the time required for the letter to arrive at a processing center, be opened, and be put in the queue to be acted upon. During the next update cycle, the removal request is effected. The actual time of removal will vary by operation. Some will update a central database on receipt and let a computer generate the necessary change orders to finish the process. Others will physically batch the paperwork and process the paperwork at a set time. Rare is the operation, using a Web page, that will update the database directly to remove the physical handling of the removal request – and such automatic removals need some mechanism to prevent a rogue removal request from taking effect.

The same process is used to effect address changes.

Mail return is handled by the United States Post Office in various ways. For first class and second class mail, the mail piece is returned to the sender, and the sender can then take corrective action: contact the recipient using another method, make an address correction if the corrected address is provided, or remove the recipient from the mailing list. There is no credit for returned mail, so there is a financial incentive to the sender to prune dead addresses from mailing lists.

A ten-day time limit for removal from such a paper operation, because of the manual labor involved, makes a great deal of sense.

REMOVAL AND FULFILLMENT IN ELECTRONIC MAIL

In order to understand the core differences between paper and electronic fulfillment, we need to look at the process of sending an electronic mail message.

A computer that understands TCP/IP protocol¹ and SMTP protocol² enters into negotiations with a remote MX (Mail Exchanger) server, which can be the collection point of mail for a recipient or a *mail relay* computer, used as an intermediate stop for the mail, to transfer the “envelope” of the electronic letter. Included in this protocol is the identification of both the sending and receiving systems, the so-called *envelope-from* mail address that identifies the ultimate source of the mail, the *envelope-to* mail address that designates the ultimate recipient's address, and information to control the transfer the content of the mail – the DATA phase. Mail can have only one envelope-from mail address, but may have multiple envelope-to addresses. (Modern MX systems limit the number of such endpoint addresses to a small number, between 25 and 100. This is to avoid letting a mail server being used to effect a *smurf*, or amplification, denial of service attack or to make spam distribution easy, particularly if the MX server has been compromised.)

¹DDN PROTOCOL HANDBOOK, NIC 50004-6, December 1985

²*Simple Mail Transfer Protocol*, Jonathan B. Postel, August 1982, Internet Request for Comment 821

Desktop personal computers have demonstrated the ability to communicate with hundreds of MX servers at the same time, using only a 28.8-kilobit/s modem connection. The reason this is possible is that the timing requirements for SMTP are very loose, measured in tens of seconds, so even the smallest personal computer can effectively communicate with MX servers.

The SMTP exchange happens in real time, but the timing requirements are such that the information can (and usually is, in the case of modern mailing system) be filled in “on the fly.” That is, the information can be changed in a database up to the time that it is needed in the mail exchange.

Mail Exchanger software, also known as *mail transfer agents* or MTAs such as Sendmail³, QMail⁴, PostFix⁵, Exim⁶, and others, work in essentially the same way. The MTA accepts mail to send from *mail user agents* or from other MTAs and places the mail, plus all addressing information, in a queue. For those endpoints that are on the same system as the MTA, the software will remove the message from the queue and perform a local delivery in a system-specific way. For those endpoints that are on another system, the MTA then starts an SMTP transfer to the next MTA in sequence and transfer the mail.

Virtually all legitimate mail user software talks with a specific MTA to send mail. For ISP customers, the MTA is provided by the ISP on a separate server to launch mail. Many Unix systems incorporate an MTA which accepts mail directly from programs (using the ubiquitous `/usr/sbin/sendmail` interface or port 25/TCP on the local host) and proceeds to transfer the mail to the remote system, perhaps through a “smart host” (server designated as the place to send all outgoing remote mail).

A legitimate bulk mailer using his or her own system will find that a capable MTA can queue thousands to hundreds of thousands of messages at a time. The MTA will then the mail out; with a larger server, hundreds or even thousands of simultaneous deliveries are possible over suitable Internet circuits such as a broadband business-class DSL line or T1. The advantage of such a system is that identifies the system that is the source, as well as the individual or organization that sourced the mail.

There is a class of software called *mailing list software* that automates the process. The user presents the message to be sent and the name of the list to apply, and the mailing list software, along with the MTA, will deliver the mail. In addition, the mailing list software will note any delivery problems and automatically remove endpoints with permanent failures, providing a report of failed endpoints for human follow-up. Some of

³<http://www.sendmail.org/>, the Sendmail Consortium, program originally written by Eric Allman

⁴<http://www.qmail.org/>, program by Dan Bernstein

⁵<http://www.postfix.org/>, a.k.a. IBM Vmailer, program by Wietse Venema at IBM's Thomas J. Watson Research Center

⁶<http://www.exim.org/>, developed at the University of Cambridge, Robinson College, Cambridge, England

the packages available are Mailman⁷ and Majordomo⁸, among others.

The mailing list software is able to do automatic deletion of dead endpoints because, like the United States Postal Service, there is the electronic analogy of returned mail, called the *bounce message*, which is either a separate message sent back to the sender, or a status code presented to the sending MTA at the end of the data phase. A bounce message consists of a machine-readable part, and a human-readable part. The machine-readable part can be easily analyzed to see if the failure is temporary or permanent, and specifically what kind of failure occurred. Section 4.2.2 on page 36 of RFC 821 shows a list of reply codes and their meaning. Reply code 550, for example, says there is no such mailbox – and many spammer programs completely ignore this reply code when they see it. (There is also 551, which is an address-change message; many people would prefer if spammerware would also ignore this code.)

The *expected* method of being able to remove oneself from an electronic mailing list is using an electronic method, such as a sending of an *unsubscribe e-mail* or clicking on a *remove-me link*. The abuse of these systems by spammers have made people leery of using these electronic methods. What some unscrupulous people will do is take the removal notice as configuration that the e-mail address in question is a “good” address, and will proceed to offer that e-mail to others, *even if they themselves honor the removal request*. For those of us who seed “spam-traps”, grafting an opt-out with the spam-trap address is guaranteed to get the spam-trap address onto spammer lists.

Which brings us to the time limit of 10 days. In a word, it's ludicrous. The opposite extreme, “instantly,” is also ludicrous. To show why, let's look at an example of a very large address system used around the world.

THE DOMAIN NAME SYSTEM AND ITS MAINTANCE

The Internet Domain Name System is used throughout the world to convert human-understandable names to the 32-bit numeric addresses used by the TCP/IP protocol to route packets from computer to computer, country to country, shore to shore, Satchel to Paige. The master phone book is maintained by the Internic (Verisign) on 13 *root* servers; this master telephone book doesn't contain every address, but instead says which phone book, er, name server to continue the search. A complete search from the root servers may take as many as ten look-ups to complete, going from name server to name server.

This layered “phone book of the Internet” currently handles 31 million names on more than one million name servers world-wide⁹... and that's just for the .com and .net

⁷<http://www.list.org/>, from GNU and the Free Software Foundation, Inc.; originally developed under the guidance of Barry Warsaw

⁸<http://www.greatcircle.com/majordomo/>, from Great Circle Associates, Inc.

⁹<http://www.icann.org/tlds/monthly-reports/com-net/verisign-200311.pdf>, *Verisign Registry*

domains. As a model for what's practical, the author believes the Internet DNS is a perfect example of what you can expect from a well-planned system of managing address information.

When a person wants to register a domain name, he or she contacts a Registrar. That Registrar then accepts the information, and payment for the term of service. The Registrar then forwards the technical transaction to Verisign (as the keeper of the root servers) for update. Verisign collects the changes to be made, and then at a set time it will update the root servers with the new information. This update is done in batch each calendar day to minimize the disruption in serving name information.

RECOMMENDATIONS FOR SPECIFIC RULE CHANGES

Given that a company that has the responsibility of 31 million names is able to provide daily update on a consistent basis – and, more importantly, do it under a budget and well below the allowed “down time” for the service, the question of whether a mailling list of millions can be maintained in a more timely manner is answered rather convincingly.

- Recommendation 1: Removal requests received electronically before 4 PM be processed no later than 8 AM the following morning, or before any mailing run is made if made after 8 AM.
- Recommendation 2: Removal requests must be available in the same way as addition requests are available. If addition requests are accepted from third parties, then removal requests must be equally accepted from third parties, and an electronic method of removal **MUST** be provided to the recipient.
- Recommendation 3: Removal requests **MUST** be accepted by United State Postal Mail, and effected no later than 8 AM the following morning after receipt, or before any mailing run is made if made after 8 AM.

TECHNOLOGY AND MARKET CHANGES THAT AFFECT CAN SPAM

This section addresses question F regarding National Do Not Email

This section addresses three issues with domain names. It used to be that a number of people would share mailbox name space in a single domain. The trend is fast going away from that:

Role-based email addresses vs. Do Not Email: There are problems with electronic mail addresses that stem from this thought problem:

Question: You have 100 departments which receive mail. What is the most cost-effective way to sort the mail?

Answer: Rent 100 post office boxes, and let the post office do the sorting. They're a lot cheaper than the cheapest mail clerk.

Businesses are finding that separate electronic mailboxes are *cheap* and do a wonderful job of sorting all the mail that comes into a business, even to the point of having separate mailboxes for each product and each phase of product marketing: sales, pre-sales support, post-sales support, warranty, and RMA, just to name a few. Modern computer MTAs can handle hundreds of thousands of mailboxes with ease, and popular mail user programs can handle tens to hundreds of mailboxes with almost as much ease.

Single-person domain names vs. Do-Not-Email: There has been an explosive growth in the number of vanity domain names that appear in the .com, .net, and .org hierarchies, as well as in the vanity-specific .name top-level domain. These are of the form <firstname><lastname>.com, such as “brianfairchild.com”, and are used by a single person. As such, a single person can have a very, very large number of possible mail names.

Catch-all mailers vs. Do-Not-Email: Businesses worrying about loss of business due to misspelled addresses are working hard to cover common misspellings. Of particular note to the discussion of Do-Not-Email is the growing number of people who will direct otherwise unusable addresses to a mailbox that is the “catch-all” for the domain.

The common thread in all three issues is that the unit of control, in many instances, needs to be the *domain* and not the *mail address*. The author has the following domains of which he is the sole user: `satchell.net` `satch-test.com` `fluent-access.com` `nodoodoo.net` `softwarr.com` `satchell.org`. **In each case, the author would like to opt out *all names* for those domains, and not have to make an exhaustive list of more than a million names for each domain – that would overburden any vendor of Do Not Email, not to mention balloon the number of CD-ROMs or even DVD-ROMs needed to distribute the list.**

Another issue with any Do Not Email list is that it becomes a piece of information, to be brokered and sold just like any other database. CAN SPAM stops at the borders

of the United States, which means that it can become very valuable to a US-based spammer with his mail operation in spammer-friendly countries. A number of people, knowing this, will not use Do Not Email for that reason.

This is especially a problem with the e-mail address of children. Toy companies, for example, could well do anything to obtain such a valuable resource as the mail addresses of their target market.

Identifying specific addresses as those of children would then become a problem. By using a single list, and requiring all mailers to use the listing service to remove children and dissenting adults, there is a chance it might work.

Unfortunately, the diff(1) utility is free, available for virtually all operating systems, and therefore useful to find out just who was removed from a list of Web and UseNet scrapings.

I submit this in the hope that someone sees values in my suggestions.

Stephen Satchell