Choicelist

A minimally intrusive, recipient configurable, authentication and permission granting system for Email

May 2003

By
John Fenley

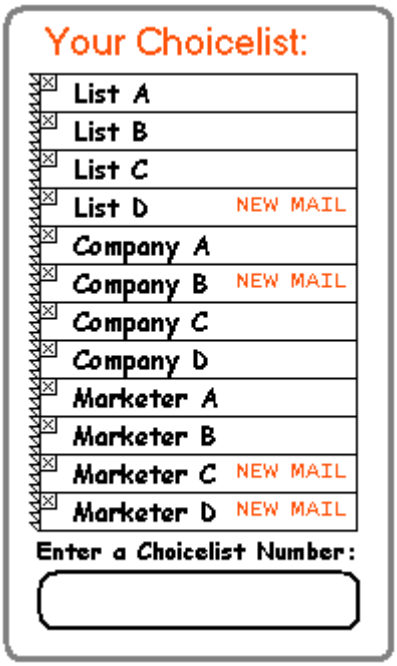
Introduction: Currently, there is no easy way for a bulk mailer to ensure that automatic mailings a recipient has asked for pass through Spam filters to reach the recipient's inbox. Also, there is no easy way for a recipient to ensure that Opt-Out requests are honored. Choicelist provides a way for these 2 things to happen easily, as well as provide other information such as content labeling, or authentication information (PgP signature key, authorized sending IP address, Etc…). Unlike other systems that have been proposed, Choicelist requires only minimal administrative action, no policy enforcement, and no changes to Email sending hardware or software. With the addition of Choicelist by Proxy, no changes are required at the receiving end either.

Choicelist

The Choicelist system was developed to solve one of the largest problems with Challenge/Response systems: Their tendency to challenge wanted, automatic mail. Choicelist is basically a directory of email addresses used to send automatic messages. But Choicelist is much more than that. The features it allows would not otherwise be possible without massive changes to email infrastructure. It is simple yet at the same time quite interdependent. The easiest way to describe how Choicelist works is to show how senders and receivers will see and use the system, then explain the way it works.

The Recipient:

The recipient interacts with the Choicelist system through a list like the one below. It lists all of the Choicelist identities they have chosen to receive mail from. Only mail from these few sources will be delivered through Choicelist.



Each source can include mail from multiple addresses, and from multiple companies, at the discretion of the identity controller.

Any mail from other Choicelist identities is blocked. This allows the user to instantly Opt-Out from further mailings from that source by simply deleting the entry from the list. No other policy enforcement is necessary, as the threat of recipient Opt-Out is enough to prevent abuse of the system by mailers.

Adding entries to the list is done by entering the Choicelist identity number of the desired list into the box provided, or clicking on a special Opt-In link such as:
<A HREF="mailto:Choicelist12345">Opt-In</A>

A link like this would be recognized by the MUA.
The user would then be asked to confirm the Opt-In.

Features impossible without Choicelist:
Instant, and 100% effective Opt-Out.
Complete list of all sources of automatic mail.
Guaranteed delivery of wanted mail

Advanced options can include the ability to access the Choicelist database in a deceptive way that prevents any party from figuring out which entries are being searched for, the ability to turn off sender notifications(stealth mode), and the ability to block even wanted messages based on content rating (for business use).

Choicelist by Proxy:

Choicelist by Proxy would be less secure and fast than using an MUA compatible with Choicelist, but would allow anyone to gain the advantages of Choicelist without changing Email software.
Basically to use Choicelist by Proxy, a user signs up at a Free, Choicelist compatible, Email service. Then forwards all mail received at their new address, to the Email address they would like to use. The Choicelist mail would all appear to come from their new address, so it should be whitelisted. The user would need to change their Choicelist preferences at the new provider, and use that new address to sign up for mailings, but the advantages of instant Opt-out, guaranteed delivery, and source listing would be gained.

Choicelist

The Sender:

Choicelist is intended to allow mass mailers to identify themselves to bypass Spam filters. Choicelist does not place any new restrictions on Email senders. There are no levels of trust, volume limits, or standards of any kind to be followed. The existing systems in use to send mail do not need to be modified in any way. Mail sent does not need to be tagged, or specially composed. And mailing lists do not need to be separated into groups based on the recipient's use of Choicelist. All that is required to become a Choicelist compatible sender, is the creation of an identity in the master Choicelist database, and the addition of a sending address.

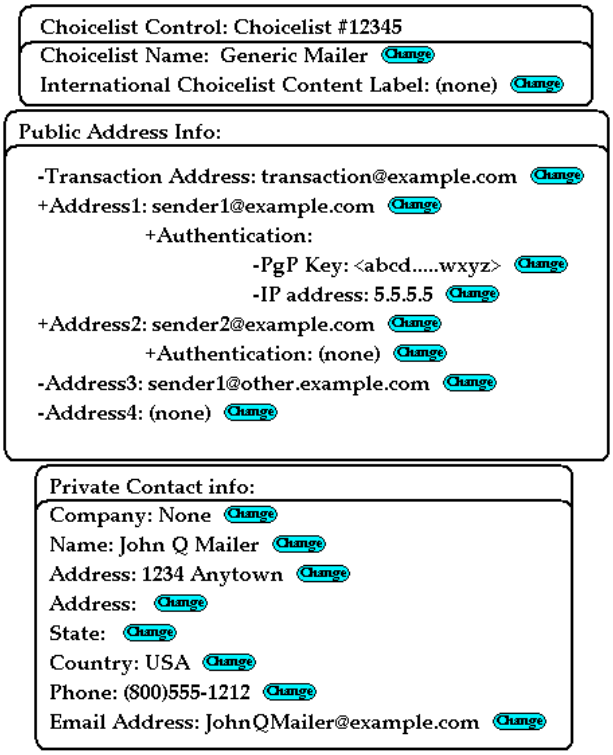
Once an entry is created, It is the list owner's responsibility to inform current subscribers of the pending Choicelist compatibility before adding the address to the identity. New subscribers should also be told about Choicelist. A special Opt-In link can be placed on websites, and the Choicelist number should be available to anyone who wants it. Again , these are just suggestions for best results, not rules to be enforced.

When an address is added to the Choicelist identity, the identity controller agrees that Choicelist systems protecting inboxes may block any mail sent by that address if the user does not have that Choicelist identity in their list.

Any number of addresses can be added to a Choicelist identity. Each address may only be in one identity at a time. Addresses may only be added with proof of ownership. Ownership will be verified by sending a message with a secret code in it to that address, then asking for that code, to complete the addition.

Once an address has been added, authentication information about that address may be added to prevent Choicelist from delivering unauthorized messages purporting to come from that address.

Mail must pass the authentication tests selected by the identity controller before it is delivered in the name of that identity.



The transaction address is an optional addition that gives an identity controller many options. Machine/Human readable messages can be sent from recipients MUAs in the following situations:

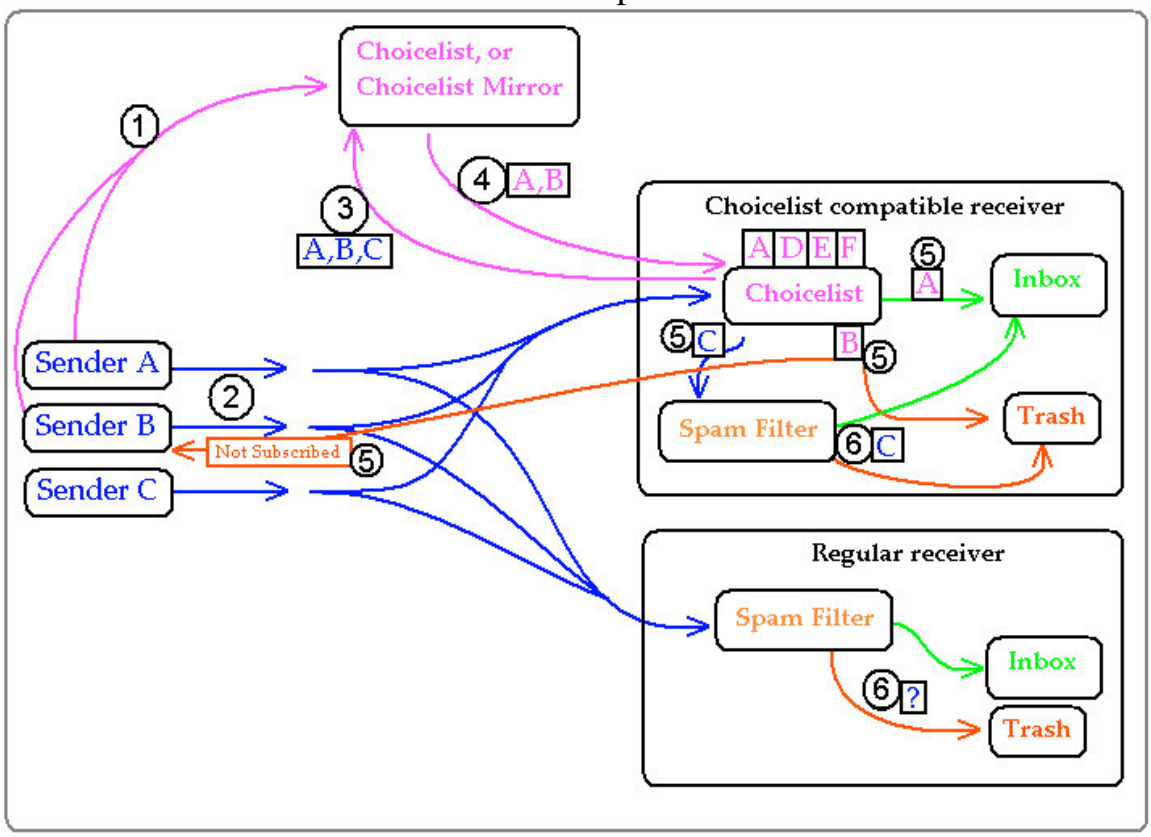
1. When a user adds an Identity to the list an Opt-In message is sent to the transaction address.
2. When a user removes an identity, an Opt-Out message is sent.
3. When a message is not delivered to a recipient, a reason such as "Failed Authentication" or "Not on personal list" is sent to help troubleshoot delivery problems, and help warn of spoofing.
4. All MUAs can be asked to "check in" to the transaction address in the case of a lost list.

Some people may not want to create a Choicelist identity if the sending address is also used as a personal address, or if the sending address simply repeats messages sent to it (E-mail discussion groups, Etc…), as the address would be made public by placement in the database. Also, Spammers would not gain an advantage by creating a Choicelist identity

Choicelist

How Choicelist Works:

An example:



1. Sender A and Sender B create Choicelist identities. Sender C does not.
2. All three senders send an Email to 2 different recipients.
3. The Choicelist recipient's MUA checks the Choicelist database for Senders A,B, and C.
4. Choicelist reports that Senders A and B have identities.
5. The message from C is sent to the Spam filter. After checking A and B against the users Choicelist. A's message is sent to the inbox, and B's message is deleted. An error message "Not Subscribed" is sent to Sender B.
6. The existing Spam filters handle the messages sent to them.

Mail received from A and B never encounter the Spam filter in the Choicelist recipient.
Mail from C is handled the same in both receivers.
Mail from all three sources is handled in the same way in the regular receiver. Choicelist does nothing to help or hinder.
The way messages are sent is irrelevant, and Choicelist can remain effective even if the protocols used to transmit them change.

Choicelist

Details:

Choicelist Mirrors: The master Choicelist database will be updated every day. The entire database will be available in weekly releases as a collection of 100 Mb files and the MD5 checksum for each one. Official daily updates will be made available as well.

Public Choicelist mirrors should respond to at least the first 2 kinds of requests through web, and direct interfaces. Mirrors should also make the source files available.
Free Choicelist server software will be made available to automate the mirroring process.
Using the public database releases to create a private mirror is the most secure way to access the database.

Standard requests: Requests to a Choicelist database can be made via Email using a standard request format. Requests, and the accompanying reply may be encrypted using public key cryptography. Requests can also be made using web forms, or Choicelist direct database access. Requests may be of 3 kinds:

1. Numerical: Returns the name, label, instructions(such as a "check in" request), the transaction address, and all valid sending addresses for an identity and the authentication information for each one.
2. Address based: Returns an identity number plus authentication information and the transaction address, or Zero, for each address requested.
3. Private(for the mildly paranoid): Requests a list of available private alphabetized chunks. From the list of chunks, a chunk is chosen that contains the wanted data. Chunks may be requested as either identity numbers, or addresses. The process may be repeated multiple times to reduce the chunks to whatever size is requested before full transmission of data.

Choicelist compatibility: Choicelist providers must be 100% compatible with the Choicelist system in order to operate using the "Choicelist" name. Partial compatibility must be disclosed fully to all users. Compatibility requirements:

1. Choicelist mail must be sorted prior to handling by any filters including domain and IP blacklists.
2. Opt-in requests from links must be confirmed by the user.
3. All forms of standard Choicelist authentication (List under construction) must be handled correctly.
4. Notifications to senders must be enabled by default. The user must be able to disable them. The user must not be given an incentive to disable them.
5. Private requests (if offered) must be disabled by default. They may be enabled later by the user(if offered).
6. Any system directly handling mail for more than a thousand individual addresses must use a mirror of the database.

Database Information: The database is basically most of the idea. It is what makes the whole thing work. The database can hold the following information:

1. The identity number. This number would be the main index in the database.
2. Contact information. This information is private, and is used only to return the Choicelist identity to the correct party in the event that an entry is hijacked.
3. Password. This password would be self encrypted.
4. A Name. This name is what a user will see on their Choicelist.
5. Instructions for subscribing MUAs
6. A standard international numeric content label (such as that proposed by the ePrivacy Group)
7. A transaction address
8. Any number of addresses may be added to the database.
9. Each address may have various amounts of authentication information.
10. The ability to add new fields of information as needed, or wanted.

Choicelist

Adoption: Choicelist provides benefits even at extremely low use levels(one sender, and one receiver), and as more people use the system, it becomes more and more advantageous to use it. The minimal amount of work required on the sender side, and Choicelist by Proxy, create an environment in which individual experimentation has very few drawbacks. Choicelist by Proxy is a service that many will use as a "throw away" address, and once the advantages are seen, they may decide to use it as their main email account as spam floods their other addresses.

At extremely high adoption rates, mirror servers will help to spread the access load to the database.

Funding: The Choicelist system does not have a revenue model. It must be free to access, and free to add to. Any cost model would slow adoption, add many new accounting issues and questions, and could prevent the system from being supported by users due to fears that the system could go under in a short while. For these reasons this plan has the maximum chance of success if it is funded and controlled by a government agency.

This system is not dissimilar to the "national do not call registry" and the new "do not Spam" registry that has been proposed. In fact, Choicelist makes a perfect compliment to the "do not Spam registry". Because Choicelist gives the user total control, a mailer need not fear sending mail to an address in the registry, because if they are both using Choicelist, any mail delivered would not be Spam.

Conclusion:

Choicelist requires that a public database be created and maintained for a substantial length of time.
Choicelist requires only a minimal amount of change on the part of senders and receivers.
Choicelist solves 3 underdeveloped needs: Sender delivery rights, authentication, and Opt-Out rights.
Choicelist allows Spam filters to concentrate on true Spam because it will handle legitimate bulk mailers.
Choicelist can remain in place unchanged if SMTP is replaced by another protocol.
Choicelist has the maximum chance of success if it is implemented under a government agency.
The system presented here is intended as a guide towards the development of a fully refined solution.

Written by:
John Fenley
John@Choicelist.com
Any comments and suggestions are welcome.
www.Choicelist.com