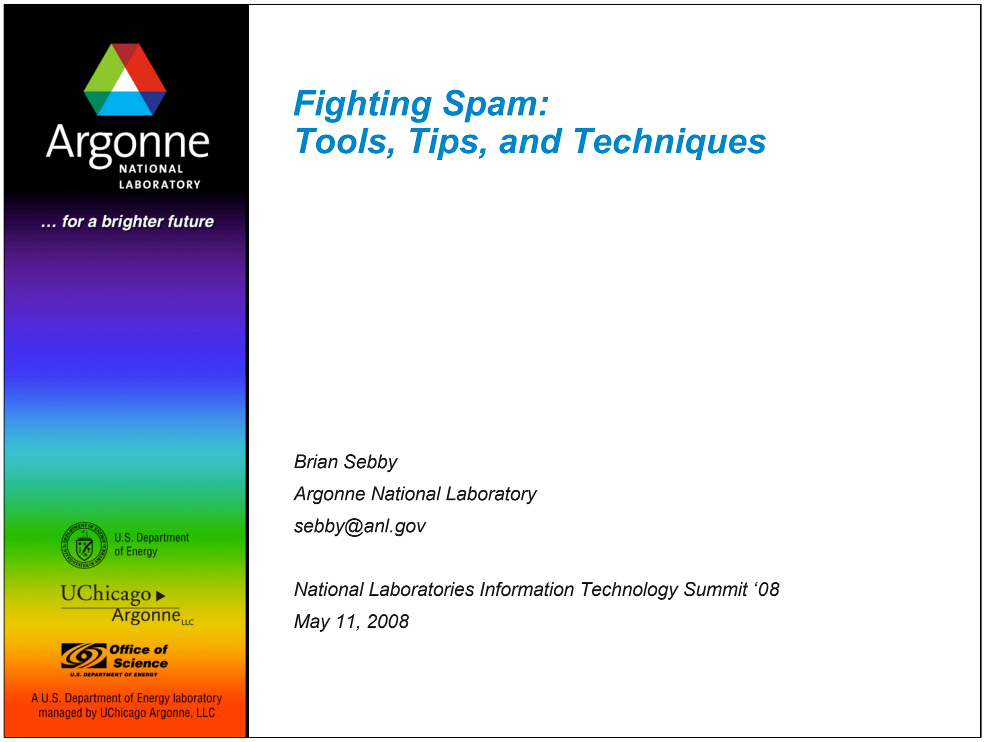


Argonne NATIONAL LABORATORY

... for a brighter future

U.S. Department of Energy

UChicago Argonne LLC

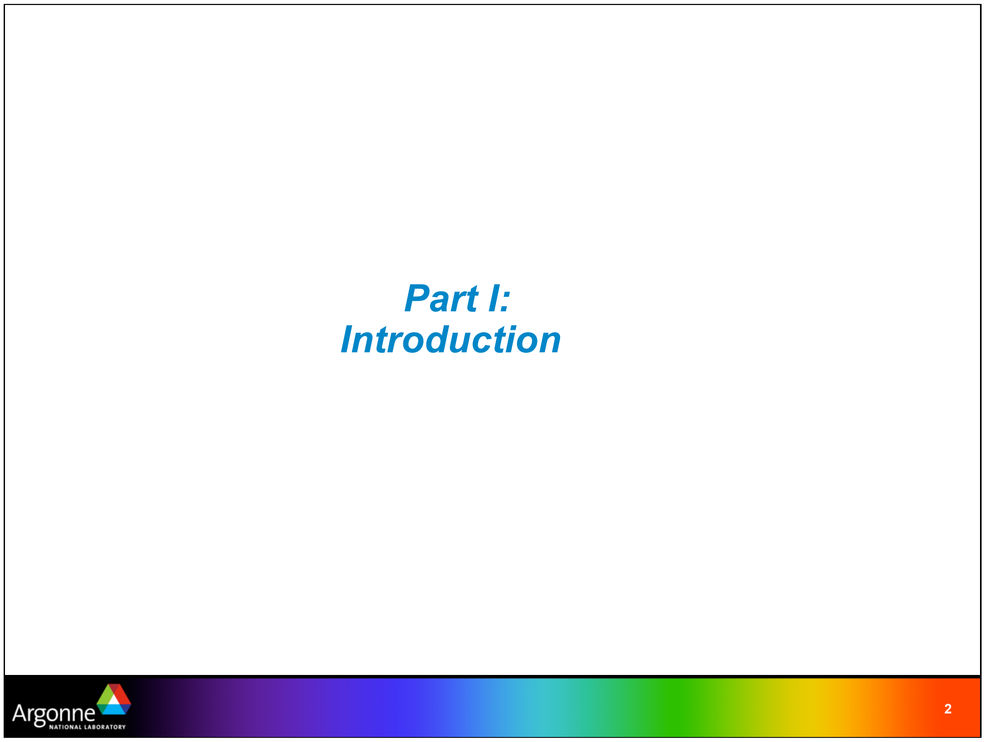
Office of Science U.S. DEPARTMENT OF ENERGY

A U.S. Department of Energy laboratory managed by UChicago Argonne, LLC

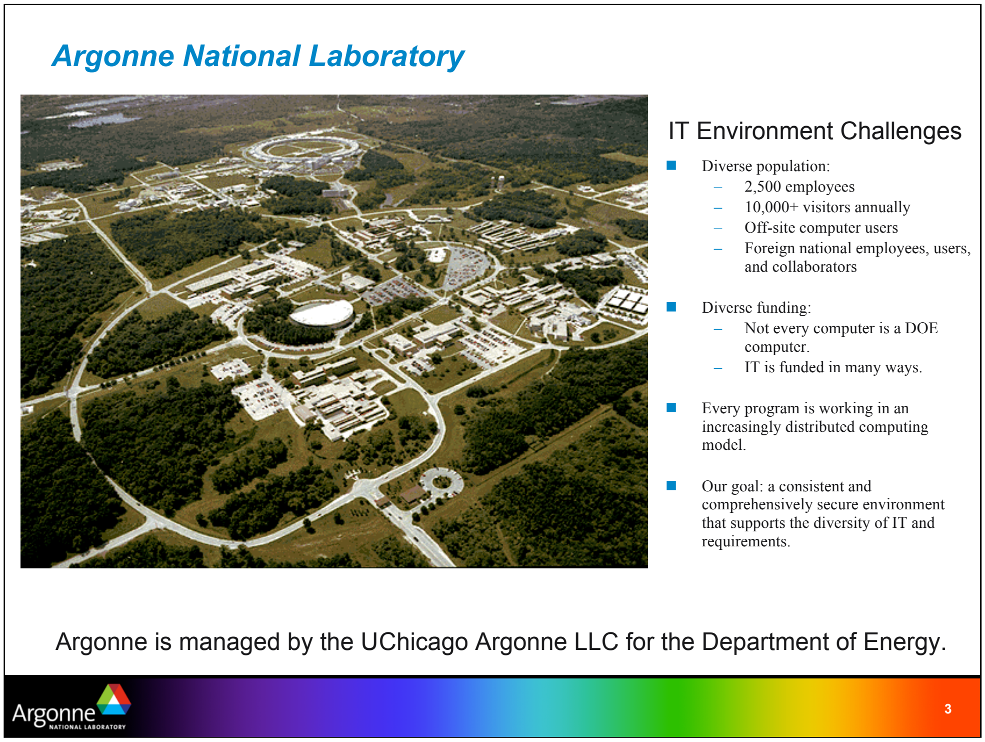
Fighting Spam: Tools, Tips, and Techniques

Brian Sebby
Argonne National Laboratory
sebby@anl.gov

National Laboratories Information Technology Summit ‘08
May 11, 2008



Part I: Introduction

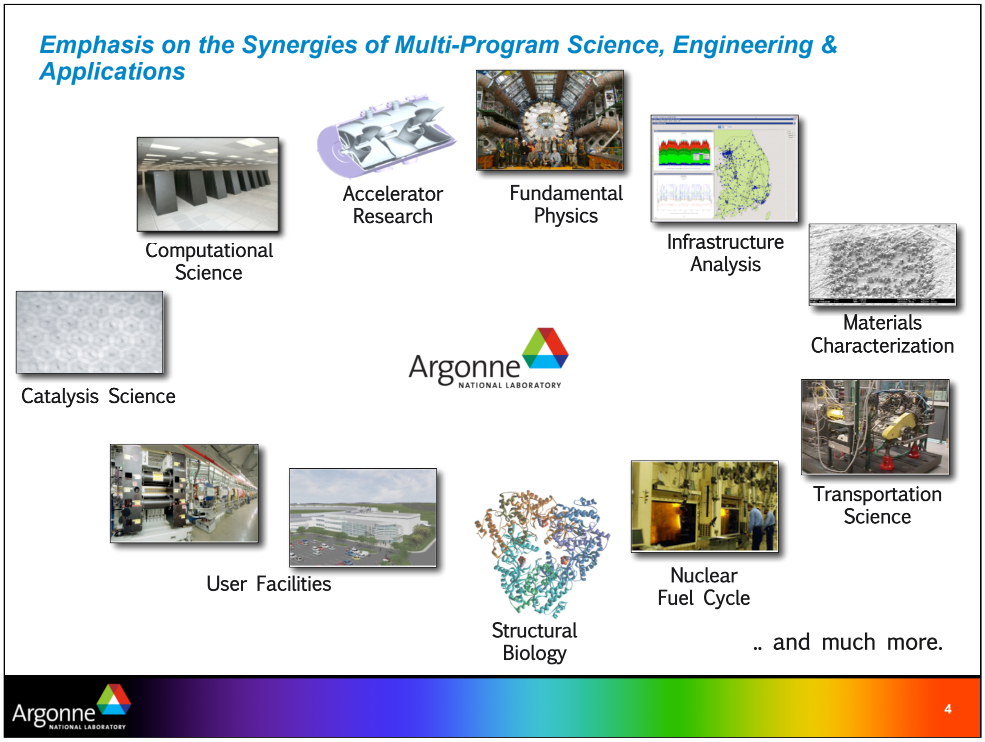


Argonne National Laboratory

IT Environment Challenges

- Diverse population:
  - 2,500 employees
  - 10,000+ visitors annually
  - Off-site computer users
  - Foreign national employees, users, and collaborators
- Diverse funding:
  - Not every computer is a DOE computer.
  - IT is funded in many ways.
- Every program is working in an increasingly distributed computing model.
- Our goal: a consistent and comprehensively secure environment that supports the diversity of IT and requirements.

Argonne is managed by the UChicago Argonne LLC for the Department of Energy.



Emphasis on the Synergies of Multi-Program Science, Engineering & Applications

Computational Science

Accelerator Research

Fundamental Physics

Infrastructure Analysis

Materials Characterization

Catalysis Science

Argonne NATIONAL LABORATORY

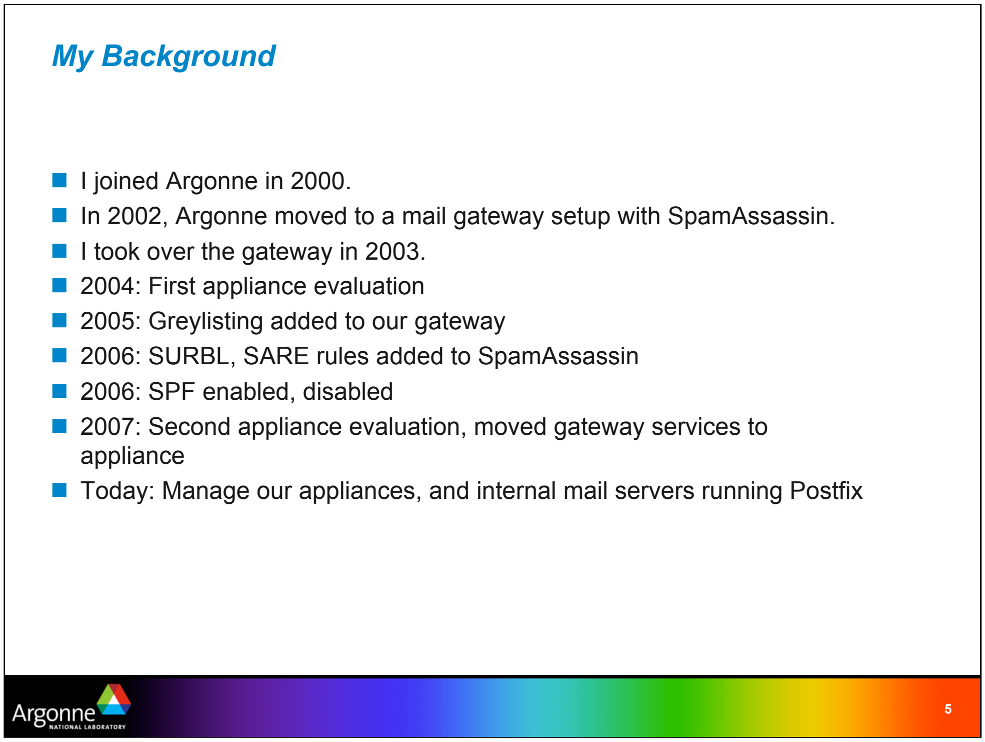
Transportation Science

User Facilities

Structural Biology

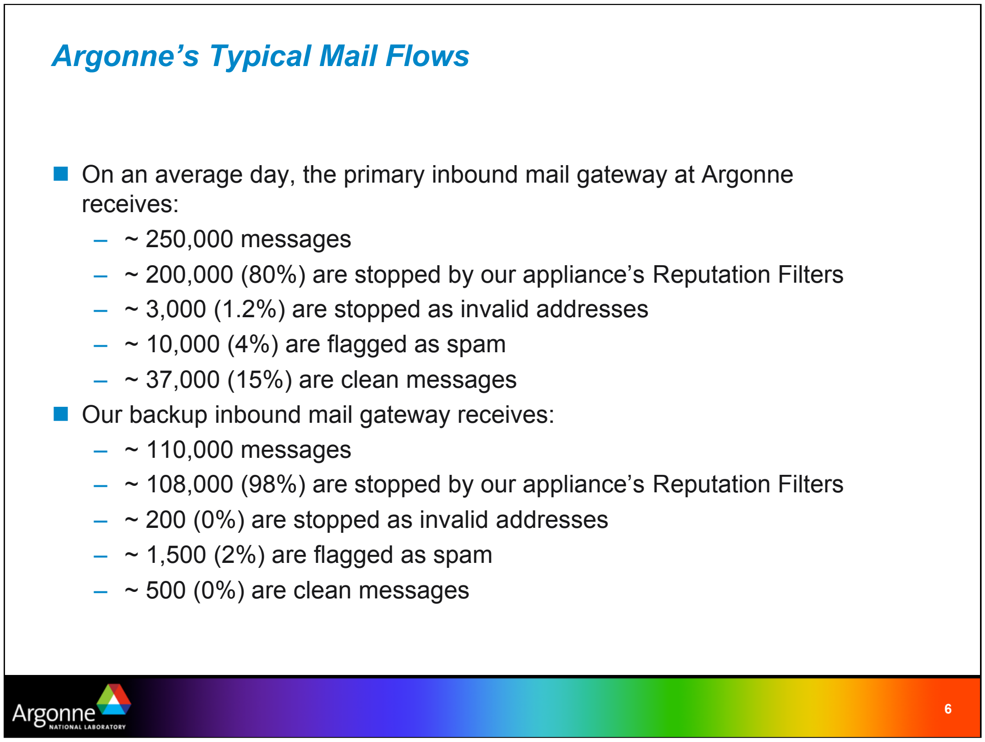
Nuclear Fuel Cycle

.. and much more.



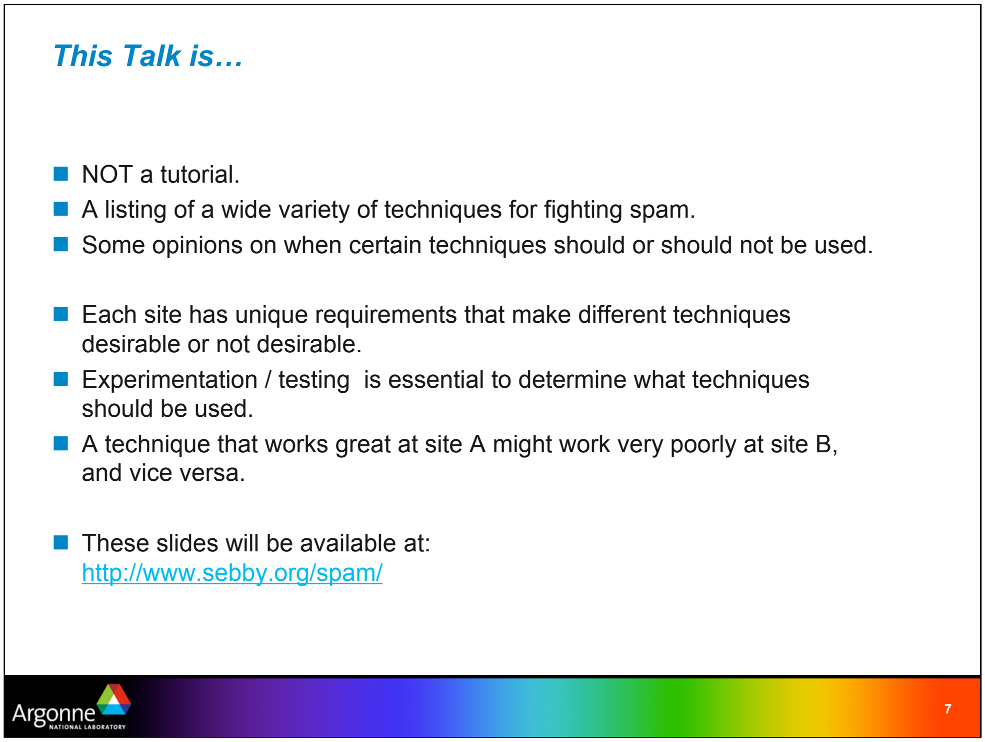
My Background

- I joined Argonne in 2000.
- In 2002, Argonne moved to a mail gateway setup with SpamAssassin.
- I took over the gateway in 2003.
- 2004: First appliance evaluation
- 2005: Greylisting added to our gateway
- 2006: SURBL, SARE rules added to SpamAssassin
- 2006: SPF enabled, disabled
- 2007: Second appliance evaluation, moved gateway services to appliance
- Today: Manage our appliances, and internal mail servers running Postfix



Argonne's Typical Mail Flows

- On an average day, the primary inbound mail gateway at Argonne receives:
  - ~ 250,000 messages
  - ~ 200,000 (80%) are stopped by our appliance's Reputation Filters
  - ~ 3,000 (1.2%) are stopped as invalid addresses
  - ~ 10,000 (4%) are flagged as spam
  - ~ 37,000 (15%) are clean messages
- Our backup inbound mail gateway receives:
  - ~ 110,000 messages
  - ~ 108,000 (98%) are stopped by our appliance's Reputation Filters
  - ~ 200 (0%) are stopped as invalid addresses
  - ~ 1,500 (2%) are flagged as spam
  - ~ 500 (0%) are clean messages

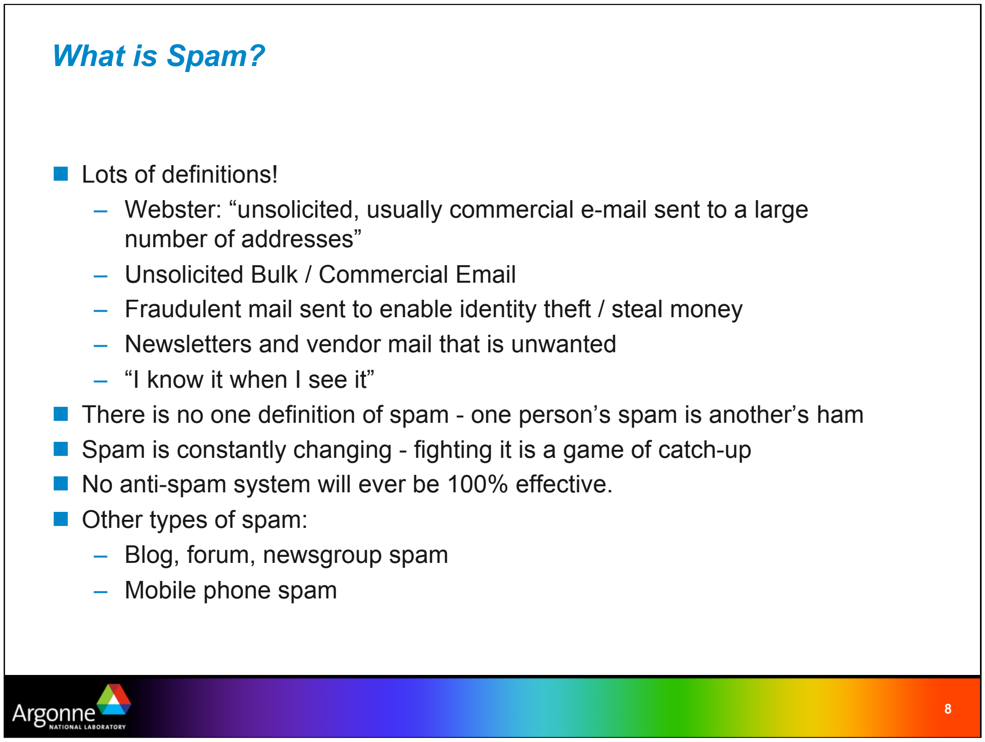


This Talk is…

- NOT a tutorial.
- A listing of a wide variety of techniques for fighting spam.
- Some opinions on when certain techniques should or should not be used.

- Each site has unique requirements that make different techniques desirable or not desirable.
- Experimentation / testing is essential to determine what techniques should be used.
- A technique that works great at site A might work very poorly at site B, and vice versa.

- These slides will be available at: http://www.sebby.org/spam/



What is Spam?

- Lots of definitions!
  - Webster: “unsolicited, usually commercial e-mail sent to a large number of addresses”
  - Unsolicited Bulk / Commercial Email
  - Fraudulent mail sent to enable identity theft / steal money
  - Newsletters and vendor mail that is unwanted
  - “I know it when I see it”
- There is no one definition of spam - one person’s spam is another’s ham
- Spam is constantly changing - fighting it is a game of catch-up
- No anti-spam system will ever be 100% effective.
- Other types of spam:
  - Blog, forum, newsgroup spam
  - Mobile phone spam

For lots more information on spam, Wikipedia and other sites have a lot of information you can read.

Some links:

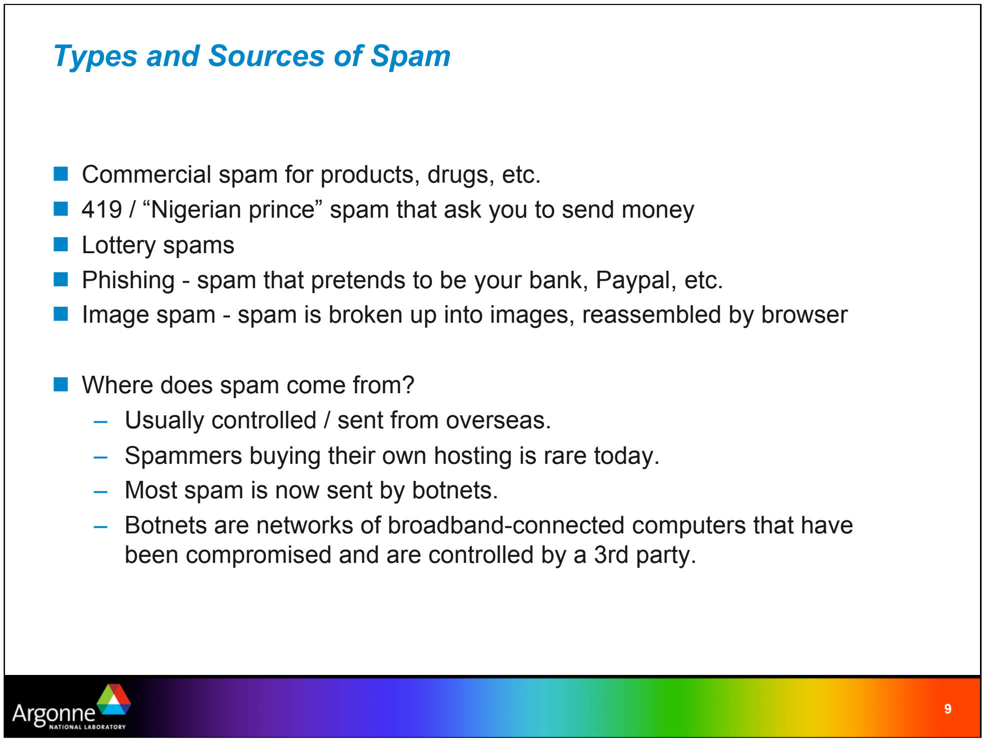
http://en.wikipedia.org/wiki/Spam_%28electronic%29

http://en.wikipedia.org/wiki/E-mail_spam

A book that might be helpful:

“Ending Spam”, Jonathan A. Zdziarski, No Starch Press, 2005

http://www.nostarch.com/endingspam.htm



Types and Sources of Spam

- Commercial spam for products, drugs, etc.
- 419 / “Nigerian prince” spam that ask you to send money
- Lottery spams
- Phishing - spam that pretends to be your bank, Paypal, etc.
- Image spam - spam is broken up into images, reassembled by browser

- Where does spam come from?
  - Usually controlled / sent from overseas.
  - Spammers buying their own hosting is rare today.
  - Most spam is now sent by botnets.
  - Botnets are networks of broadband-connected computers that have been compromised and are controlled by a 3rd party.

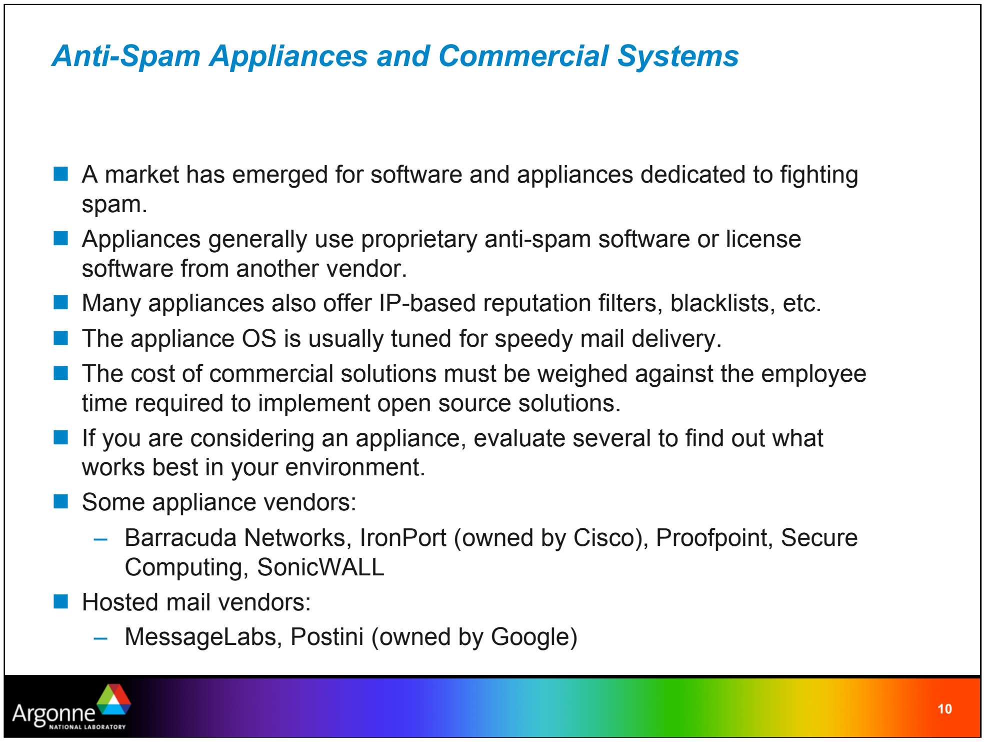
Botnets are beyond the scope of this talk, but some interesting links are:

The Honeynet Project - a group that uses honeypots (deliberately vulnerable computers) to find out what the botnets are doing.

http://www.honeynet.org/

The Shadowserver Foundation - tracks botnets and cybercrime in general.

http://www.shadowserver.org/wiki/



Anti-Spam Appliances and Commercial Systems

- A market has emerged for software and appliances dedicated to fighting spam.
- Appliances generally use proprietary anti-spam software or license software from another vendor.
- Many appliances also offer IP-based reputation filters, blacklists, etc.
- The appliance OS is usually tuned for speedy mail delivery.
- The cost of commercial solutions must be weighed against the employee time required to implement open source solutions.
- If you are considering an appliance, evaluate several to find out what works best in your environment.
- Some appliance vendors:
  - Barracuda Networks, IronPort (owned by Cisco), Proofpoint, Secure Computing, SonicWALL
- Hosted mail vendors:
  - MessageLabs, Postini (owned by Google)

Here are some URLs for vendors mentioned above.

Appliances / Commercial anti-spam software:

Barracuda Networks - http://www.barracudanetworks.com/

IronPort - http://www.ironport.com/ (Owned by Cisco Systems)

Proofpoint - http://www.proofpoint.com/

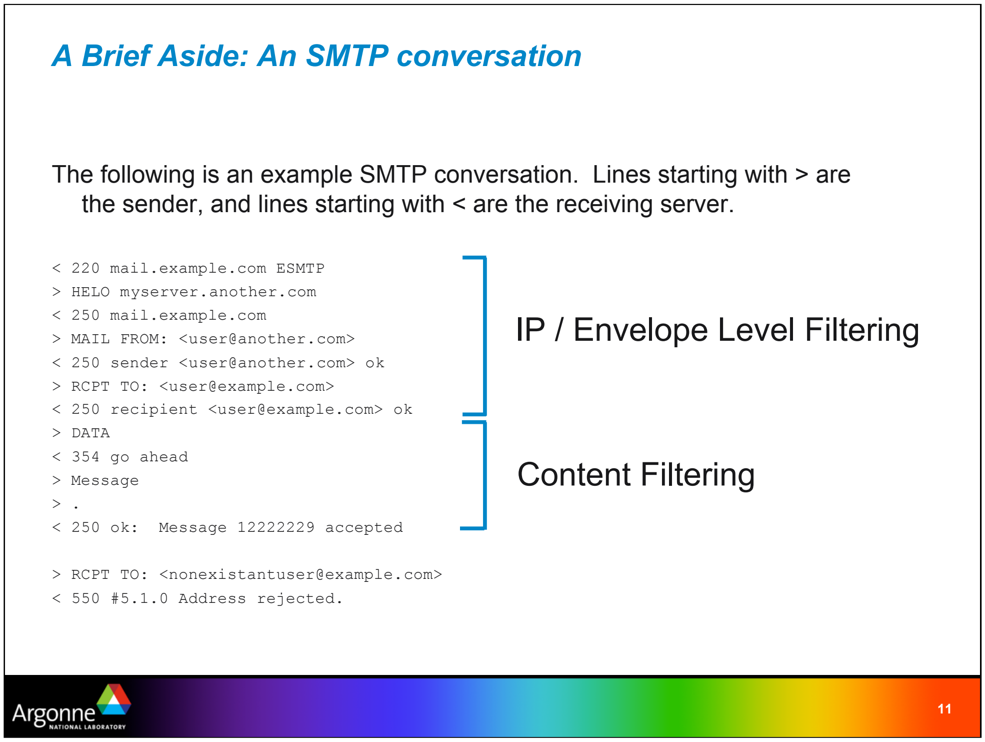
Secure Computing IronMail - http://www.securecomputing.com/

SonicWALL - http://www.sonicwall.com/

Hosted mail solutions (where you set your MX records to point at their servers):

MessageLabs - http://www.messagelabs.com/

Postini - http://www.postini.com (Owned by Google)



A Brief Aside: An SMTP conversation

The following is an example SMTP conversation. Lines starting with > are the sender, and lines starting with < are the receiving server.

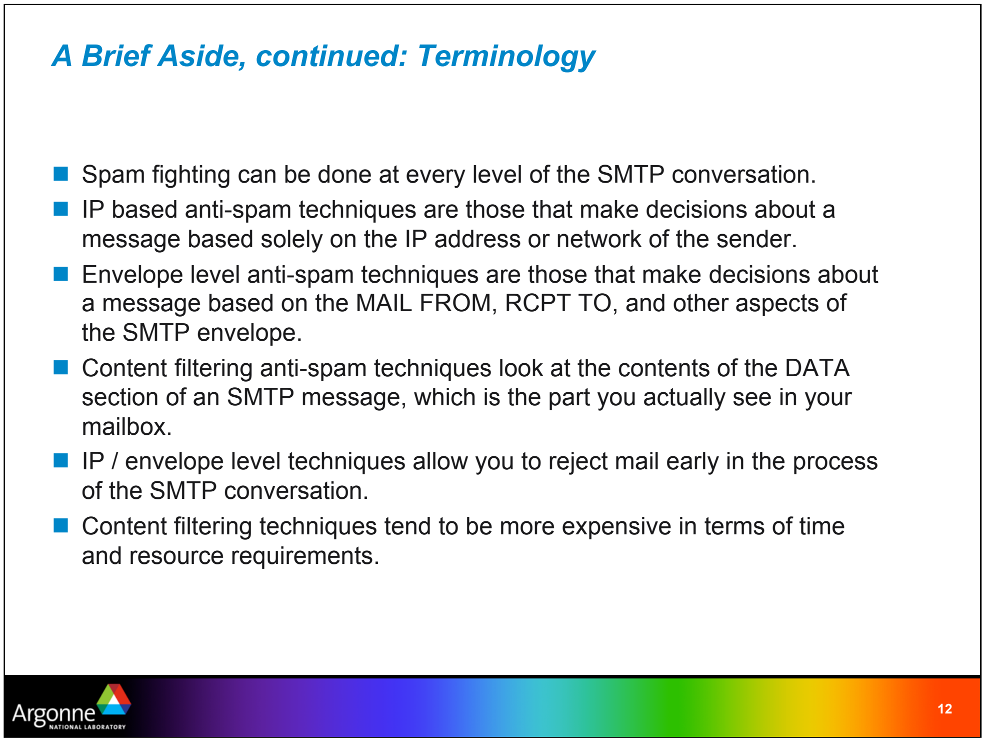
```
< 220 mail.example.com ESMTP
> HELO myserver.another.com
< 250 mail.example.com
> MAIL FROM: <user@another.com>
< 250 sender <user@another.com> ok
> RCPT TO: <user@example.com>
< 250 recipient <user@example.com> ok
> DATA
< 354 go ahead
> Message
> .
< 250 ok:  Message 12222229 accepted

> RCPT TO: <nonexistantuser@example.com>
< 550 #5.1.0 Address rejected.
```

IP / Envelope Level Filtering

Content Filtering

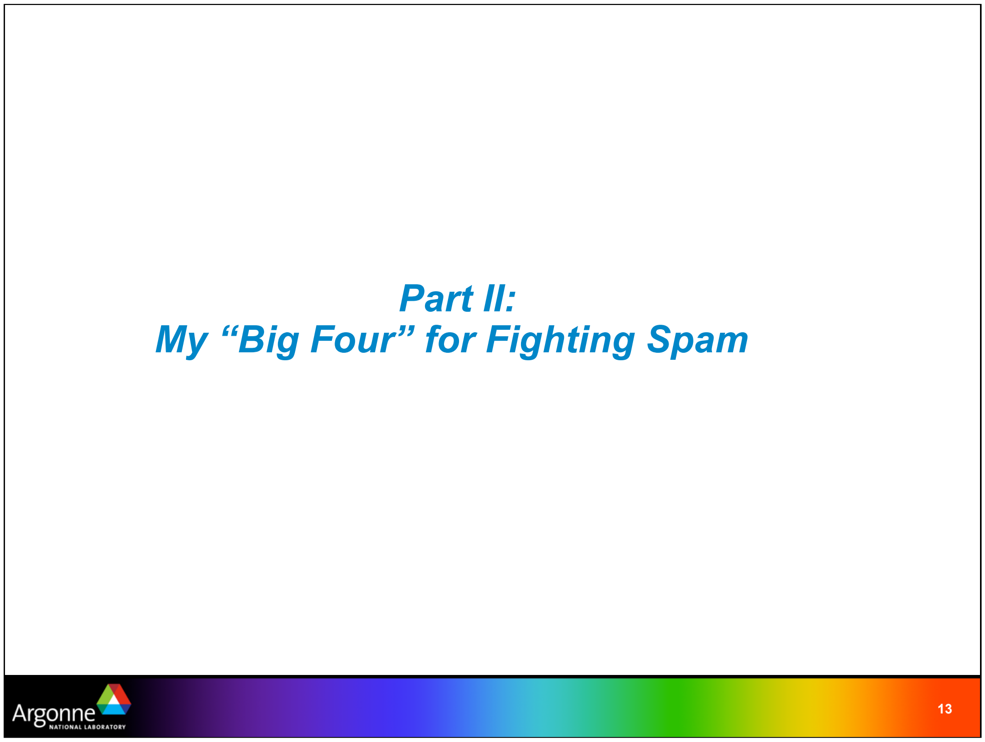
More information on SMTP can be found at:

http://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol
RFC 2821 (ESMTP envelope) - http://tools.ietf.org/html/rfc2821
RFC 2822 (Mail message format) - http://tools.ietf.org/html/rfc2822

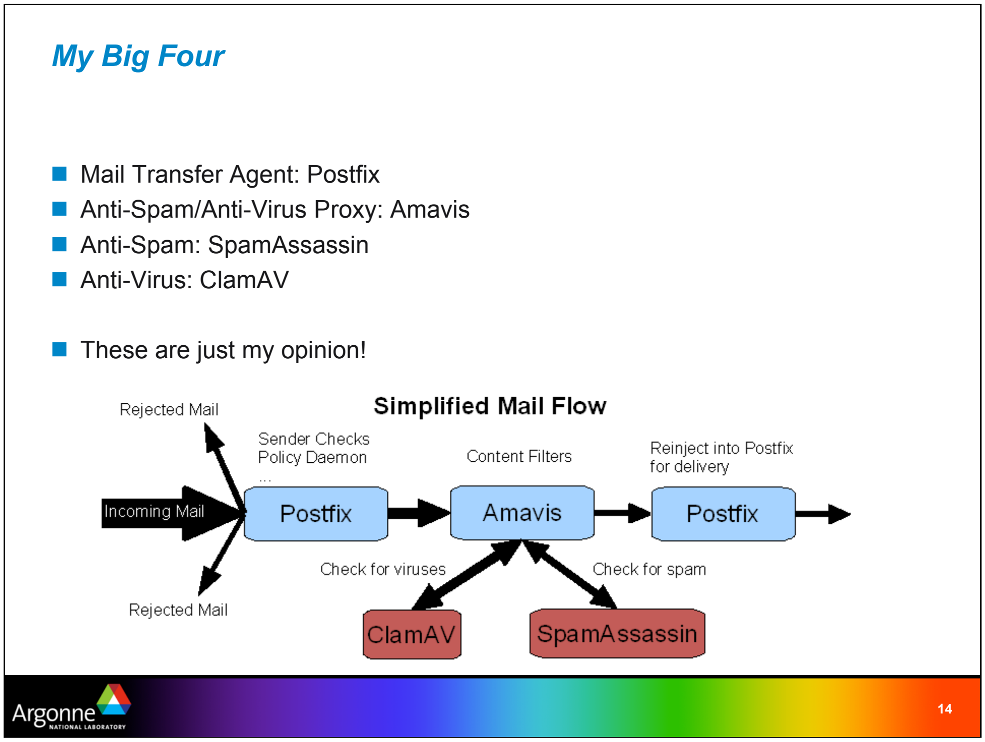


A Brief Aside, continued: Terminology

- Spam fighting can be done at every level of the SMTP conversation.
- IP based anti-spam techniques are those that make decisions about a message based solely on the IP address or network of the sender.
- Envelope level anti-spam techniques are those that make decisions about a message based on the MAIL FROM, RCPT TO, and other aspects of the SMTP envelope.
- Content filtering anti-spam techniques look at the contents of the DATA section of an SMTP message, which is the part you actually see in your mailbox.
- IP / envelope level techniques allow you to reject mail early in the process of the SMTP conversation.
- Content filtering techniques tend to be more expensive in terms of time and resource requirements.



Part II:
My “Big Four” for Fighting Spam



My Big Four

- Mail Transfer Agent: Postfix
- Anti-Spam/Anti-Virus Proxy: Amavis
- Anti-Spam: SpamAssassin
- Anti-Virus: ClamAV

- These are just my opinion!

Simplified Mail Flow

Rejected Mail

Sender Checks
Policy Daemon
...

Content Filters

Reinject into Postfix
for delivery

Incoming Mail

Postfix

Amavis

Postfix

Check for viruses

Check for spam

Rejected Mail

ClamAV

SpamAssassin

Argonne NATIONAL LABORATORY

14

The choice of an MTA tends to bring out strong emotions and opinions. My experience has been exclusively with Sendmail and Postfix. Sendmail is extremely powerful, but I found it a lot more difficult to use - thus I prefer the simplicity of Postfix.

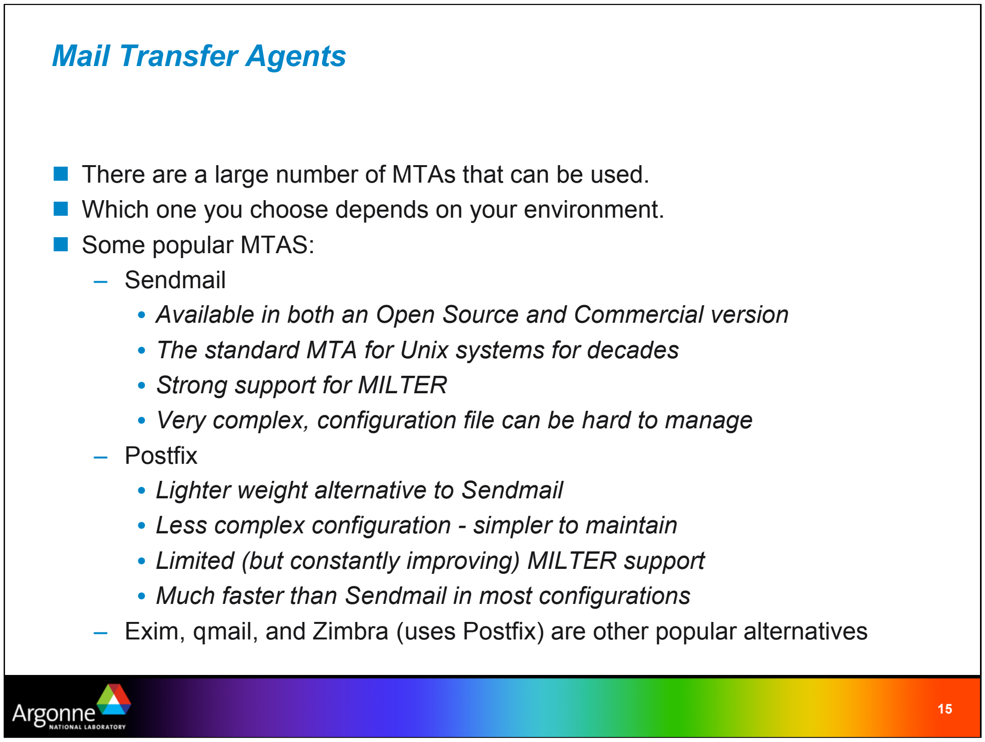
My favorite example for showing the differences between the configuration of these MTAs is how you integrate Procmail into each.

For sendmail, the following must be put in sendmail.cf:

```
Mlocal, P=/usr/bin/procmail, F=SAw5:|/@glDFMPhsfn, S=10/30,
R=20/40,
        T=DNS/RFC822/X-Unix,
        A=procmail -Y -a $h -d $u
```

For postfix, the following must be put in main.cf:

```
mailbox_command = /usr/bin/procmail
```

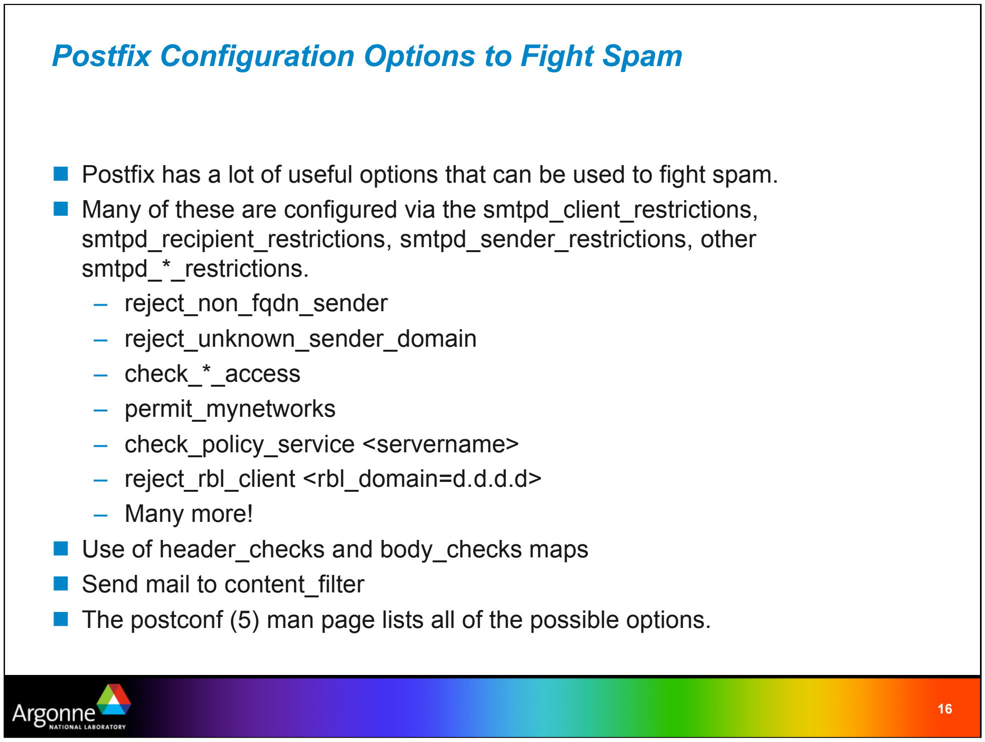


Mail Transfer Agents

- There are a large number of MTAs that can be used.
- Which one you choose depends on your environment.
- Some popular MTAS:
  - Sendmail
    - *Available in both an Open Source and Commercial version*
    - *The standard MTA for Unix systems for decades*
    - *Strong support for MILTER*
    - *Very complex, configuration file can be hard to manage*
  - Postfix
    - *Lighter weight alternative to Sendmail*
    - *Less complex configuration - simpler to maintain*
    - *Limited (but constantly improving) MILTER support*
    - *Much faster than Sendmail in most configurations*
  - Exim, qmail, and Zimbra (uses Postfix) are other popular alternatives

Here are URLs for all of these MTAs:

Postfix - http://www.postfix.org/
Sendmail - http://www.sendmail.org/ (open source version)
 - http://www.sendmail.com/ (commercial version)
Exim - http://www.exim.org/
qmail - http://cr.yp.to/qmail.html
Zimbra - http://www.zimbra.com/
Others - http://en.wikipedia.org/wiki/List_of_mail_servers



Postfix Configuration Options to Fight Spam

- Postfix has a lot of useful options that can be used to fight spam.
- Many of these are configured via the smtpd_client_restrictions, smtpd_recipient_restrictions, smtpd_sender_restrictions, other smtpd_*_restrictions.
  - reject_non_fqdn_sender
  - reject_unknown_sender_domain
  - check_*_access
  - permit_mynetworks
  - check_policy_service <servername>
  - reject_rbl_client <rbl_domain=d.d.d.d>
  - Many more!
- Use of header_checks and body_checks maps
- Send mail to content_filter
- The postconf (5) man page lists all of the possible options.

All of these options can be found at:

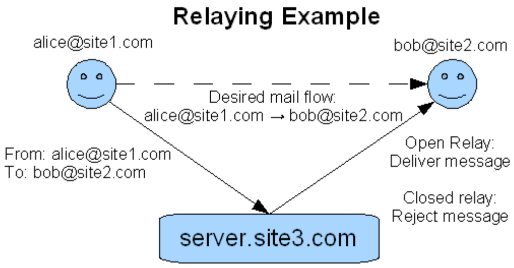
http://www.postfix.org/postconf.5.html

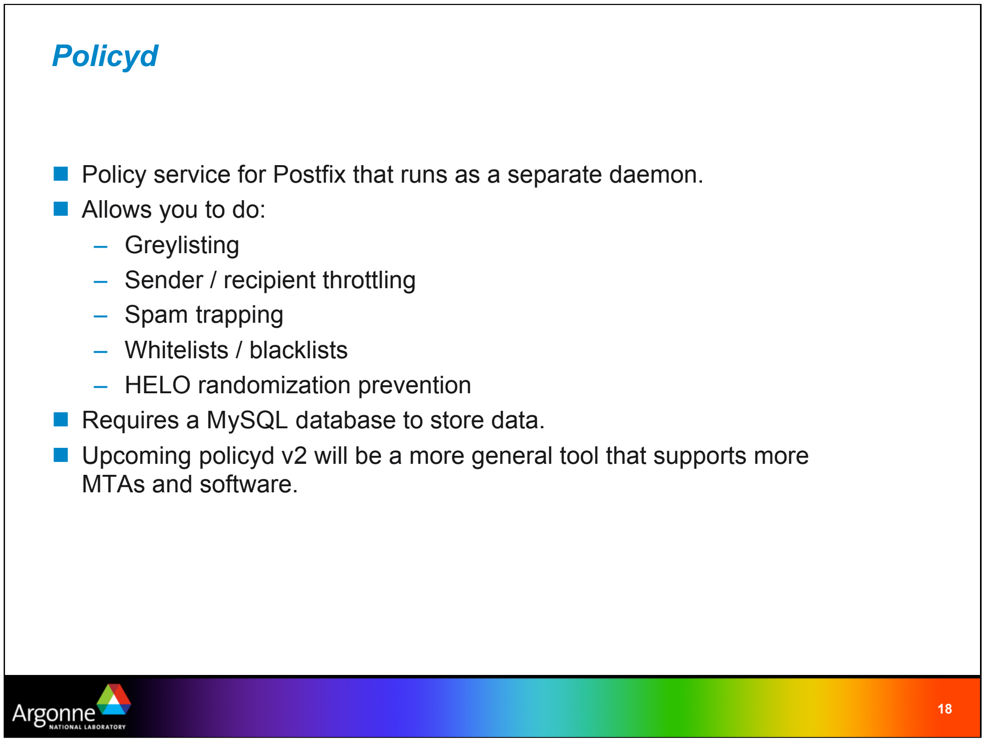
Another great resource is “The Book of Postfix” by Ralf Hildebrandt and Patrick Koetter.

http://www.nostarch.com/postfix.htm

Open Mail Relays

- A mail server is considered a relay when it accepts mail that is destined for a non-local address.
- An open mail relay is one that accepts and delivers mail from anywhere that is to be delivered to a non-local address.
- Open mail relays are used by spammers and will get your server blacklisted.
- Only allow relaying from your local network, or require authentication.
- In Postfix, use the permit_mynetworks parameter to the smtpd_*_restrictions.
- The relay_domains parameter also controls what domains you accept mail for.



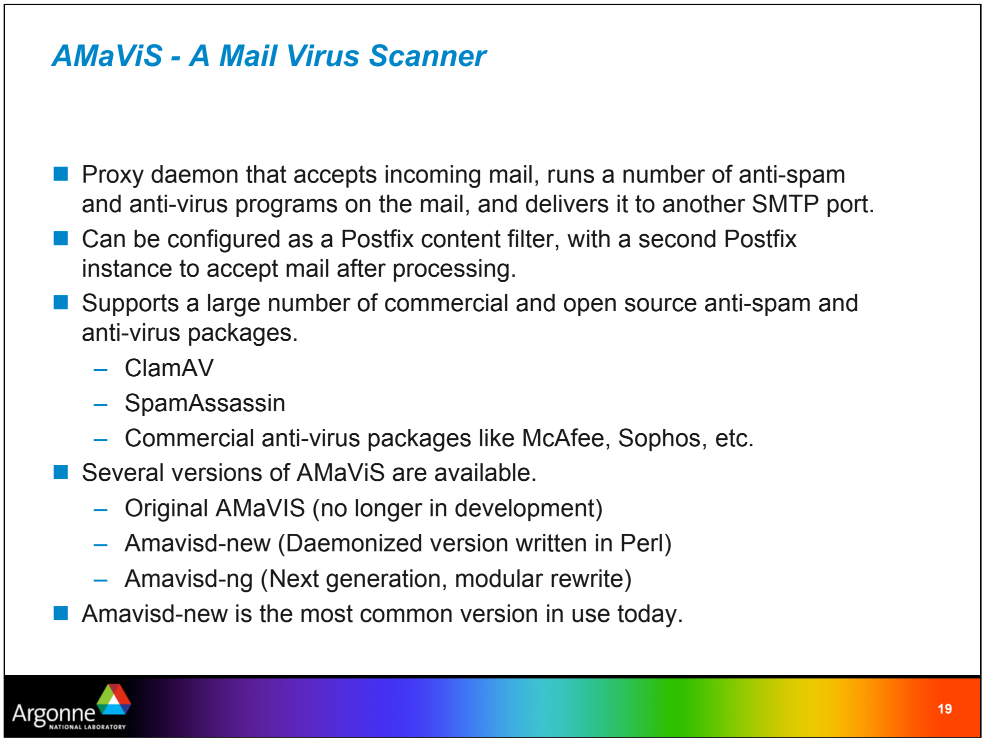


Policyd

- Policy service for Postfix that runs as a separate daemon.
- Allows you to do:
  - Greylisting
  - Sender / recipient throttling
  - Spam trapping
  - Whitelists / blacklists
  - HELO randomization prevention
- Requires a MySQL database to store data.
- Upcoming policyd v2 will be a more general tool that supports more MTAs and software.

The policyd webpage is at:

http://www.policyd.org/



AMaViS - A Mail Virus Scanner

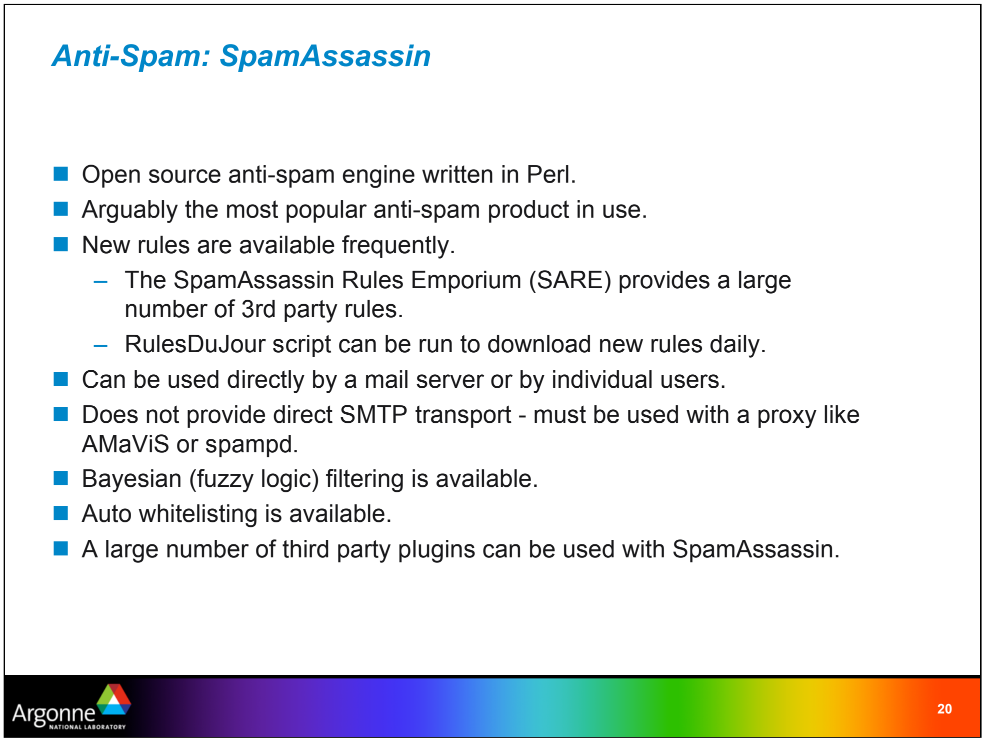
- Proxy daemon that accepts incoming mail, runs a number of anti-spam and anti-virus programs on the mail, and delivers it to another SMTP port.
- Can be configured as a Postfix content filter, with a second Postfix instance to accept mail after processing.
- Supports a large number of commercial and open source anti-spam and anti-virus packages.
  - ClamAV
  - SpamAssassin
  - Commercial anti-virus packages like McAfee, Sophos, etc.
- Several versions of AMaViS are available.
  - Original AMaVIS (no longer in development)
  - Amavisd-new (Daemonized version written in Perl)
  - Amavisd-ng (Next generation, modular rewrite)
- Amavisd-new is the most common version in use today.

Here are links to the versions of AMaViS mentioned above:

Amavis - http://www.amavis.org/

Amavisd-new - http://www.ijs.si/software/amavisd/

Amavisd-ng - http://sourceforge.net/projects/amavis



Anti-Spam: SpamAssassin

- Open source anti-spam engine written in Perl.
- Arguably the most popular anti-spam product in use.
- New rules are available frequently.
  - The SpamAssassin Rules Emporium (SARE) provides a large number of 3rd party rules.
  - RulesDuJour script can be run to download new rules daily.
- Can be used directly by a mail server or by individual users.
- Does not provide direct SMTP transport - must be used with a proxy like AMaViS or spampd.
- Bayesian (fuzzy logic) filtering is available.
- Auto whitelisting is available.
- A large number of third party plugins can be used with SpamAssassin.

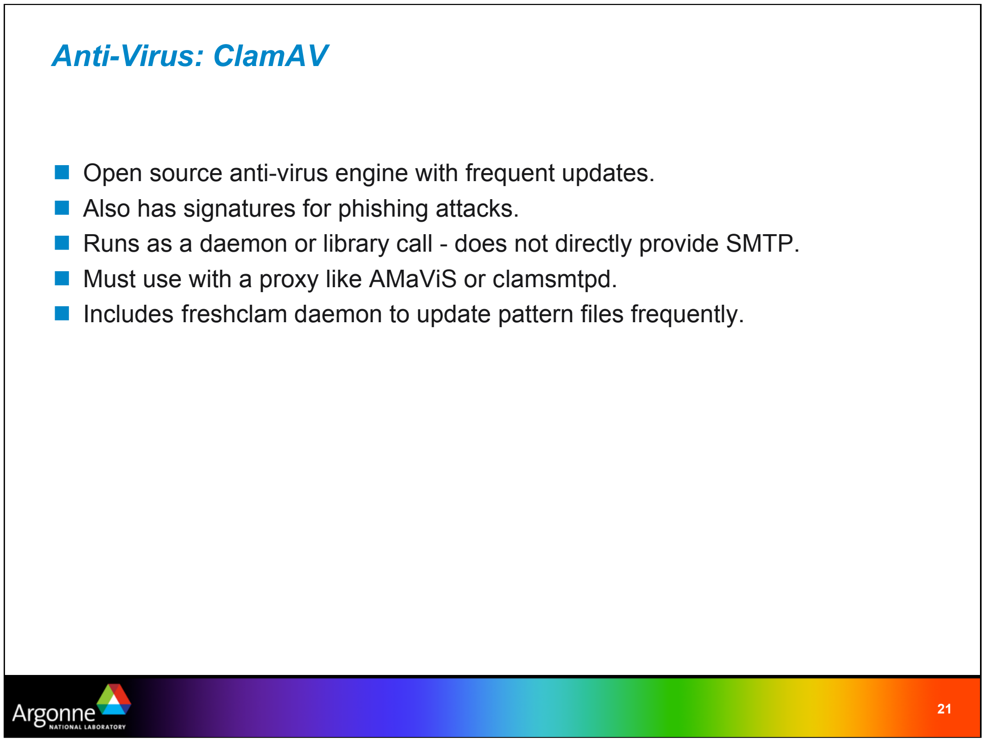
The SpamAssassin homepage is:

http://spamassassin.apache.org/

The SARE rules are available at:

http://www.rulesemporium.com/

SpamPD:
http://www.worlddesign.com/Content/rd/mta/spampd/spampd.html



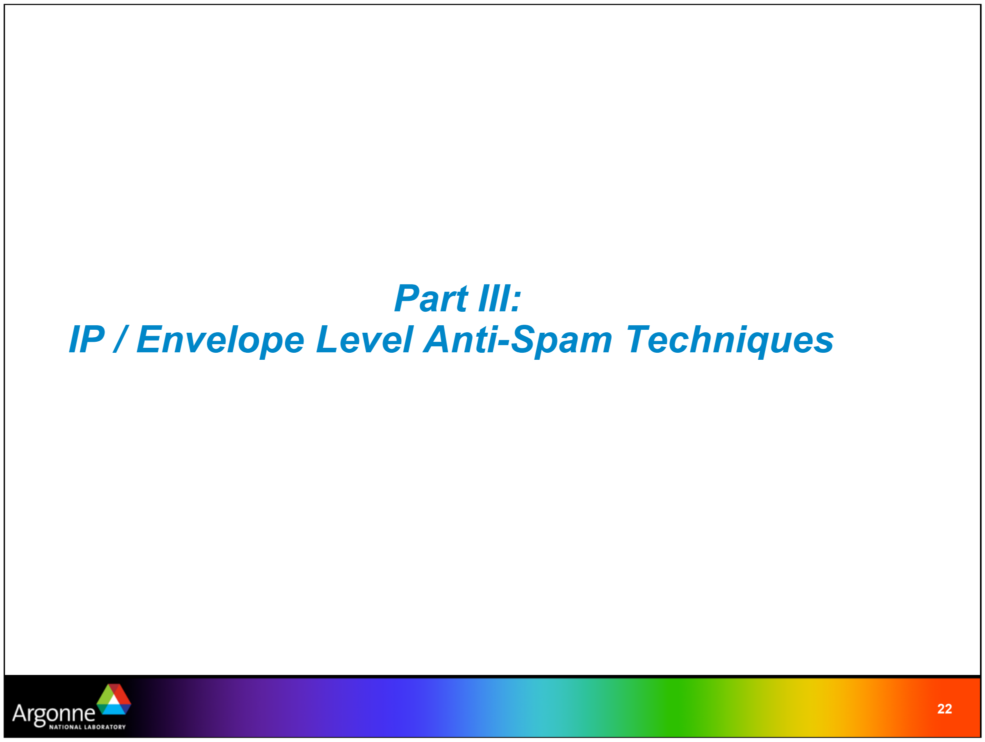
Anti-Virus: ClamAV

- Open source anti-virus engine with frequent updates.
- Also has signatures for phishing attacks.
- Runs as a daemon or library call - does not directly provide SMTP.
- Must use with a proxy like AMaViS or clamsmtpd.
- Includes freshclam daemon to update pattern files frequently.

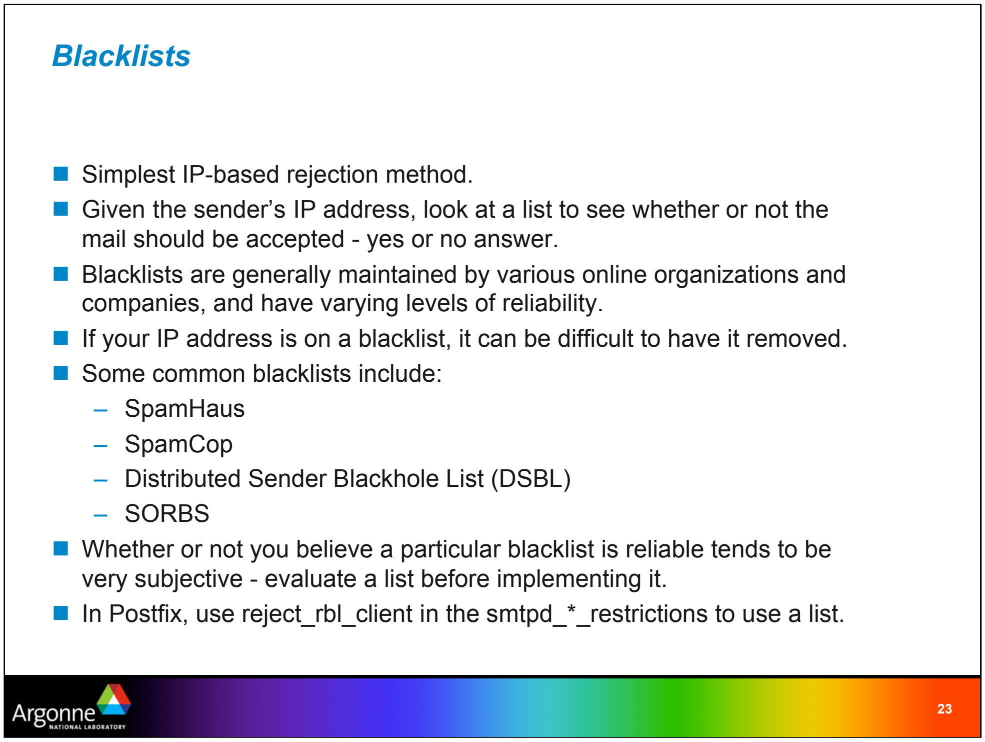
The ClamAV home page is at:

ClamAV - http://www.clamav.net/

ClamSMTP: http://memberwebs.com/stef/software/clamsmtp/



Part III:
IP / Envelope Level Anti-Spam Techniques



Blacklists

- Simplest IP-based rejection method.
- Given the sender's IP address, look at a list to see whether or not the mail should be accepted - yes or no answer.
- Blacklists are generally maintained by various online organizations and companies, and have varying levels of reliability.
- If your IP address is on a blacklist, it can be difficult to have it removed.
- Some common blacklists include:
  - SpamHaus
  - SpamCop
  - Distributed Sender Blackhole List (DSBL)
  - SORBS
- Whether or not you believe a particular blacklist is reliable tends to be very subjective - evaluate a list before implementing it.
- In Postfix, use reject_rbl_client in the smtpd_*_restrictions to use a list.

The use of blacklists and the reliability of each list is one of the most contentious areas in spam fighting - people have lists that they trust, and don't trust, and these opinions can be extremely varied. My experience with blacklists is limited, as we elected not to use them, but the ones that I have heard positive comments about are zen.spamhaus.net, list.dsbl.org, and dul.dnsbl.sorbs.net. However, like every technique, evaluate the blacklist before putting it into production.

URLs for the lists mentioned above, and a couple of links to general information about blacklists are:

DNS Blacklists - http://en.wikipedia.org/wiki/DNSBL

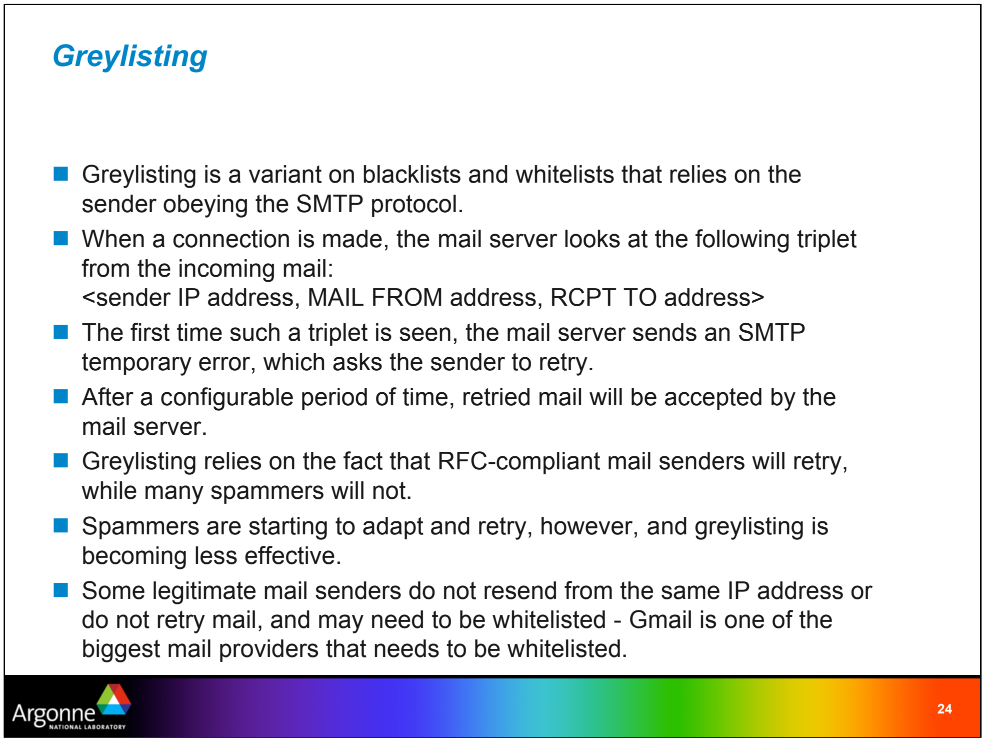
- http://en.wikipedia.org/wiki/Comparison_of_DNS_blacklists

SpamHaus - http://www.spamhaus.org/

SpamCop - http://www.spamcop.net/

DSBL - http://dsbl.org/main

SORBS - http://www.us.sorbs.net/ (controversial)



Greylisting

- Greylisting is a variant on blacklists and whitelists that relies on the sender obeying the SMTP protocol.
- When a connection is made, the mail server looks at the following triplet from the incoming mail:
  <sender IP address, MAIL FROM address, RCPT TO address>
- The first time such a triplet is seen, the mail server sends an SMTP temporary error, which asks the sender to retry.
- After a configurable period of time, retried mail will be accepted by the mail server.
- Greylisting relies on the fact that RFC-compliant mail senders will retry, while many spammers will not.
- Spammers are starting to adapt and retry, however, and greylisting is becoming less effective.
- Some legitimate mail senders do not resend from the same IP address or do not retry mail, and may need to be whitelisted - Gmail is one of the biggest mail providers that needs to be whitelisted.

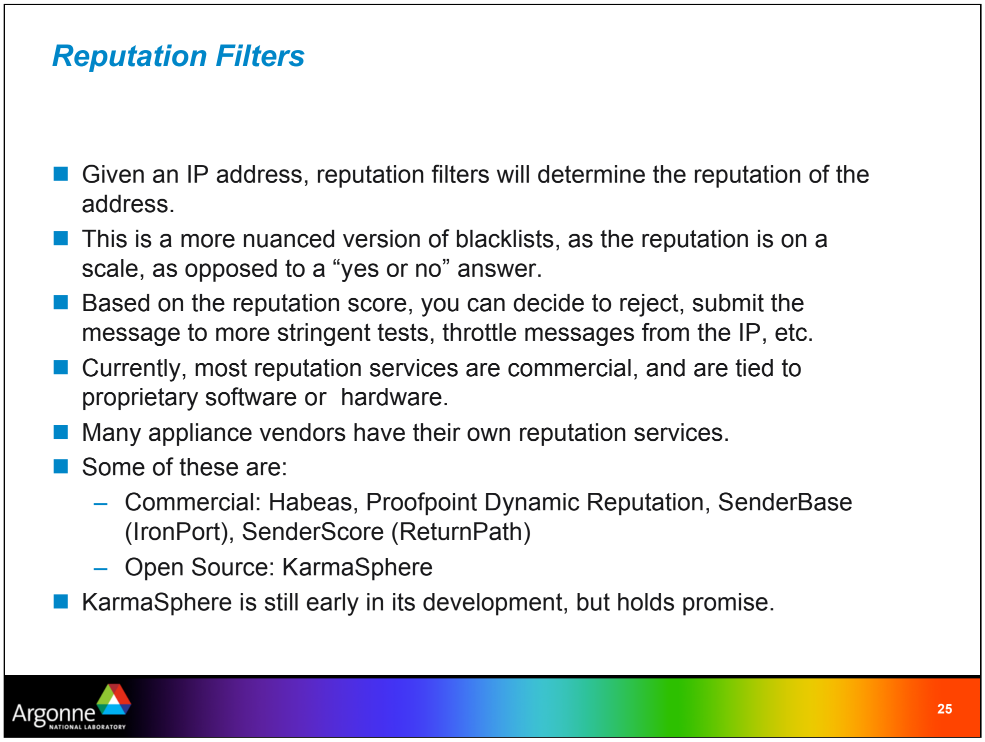
Greylisting is another somewhat controversial technique. When we first implemented greylisting, it eliminated about 40% of our incoming mail connections, and made viral attachments almost nonexistent. However, more recent reports have indicated that at some sites, the amount of mail blocked by greylisting has gone from 20% to 12% or lower as spammers are now retrying more often.

Postfix has a built-in example greylisting policy service, and the postgrey package is available to provide a more robust solution. Postgrey uses a Berkeley DB to store the list of triplets that it has already seen.

URLs for more information are:

Greylisting - http://www.greylisting.org/

Postgrey - http://postgrey.schweikert.ch/



Reputation Filters

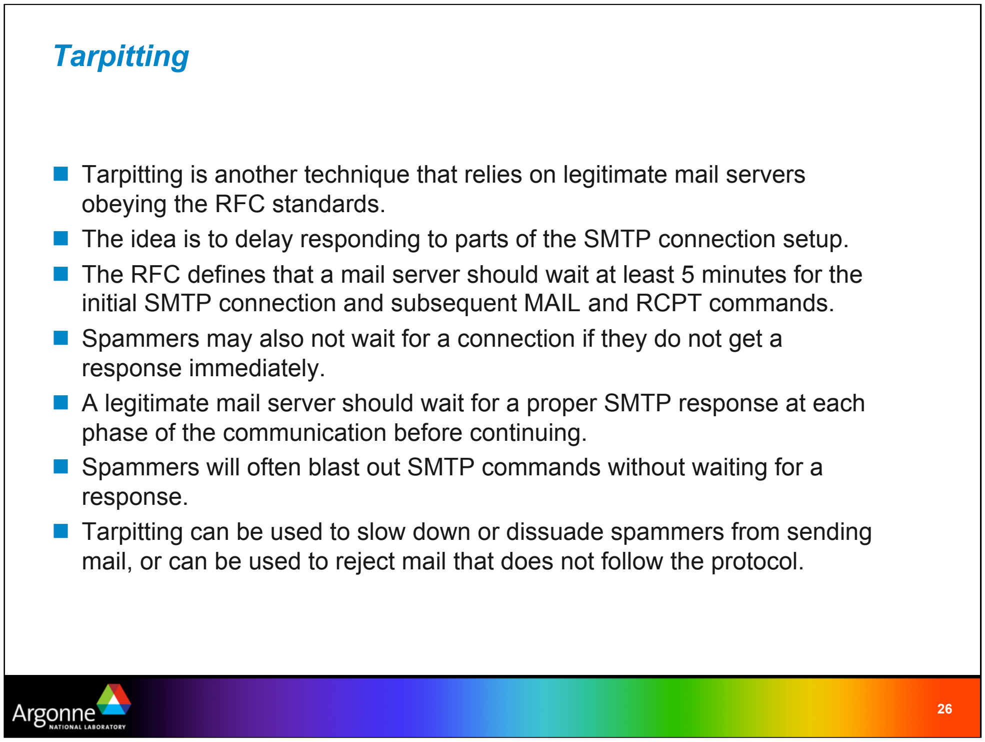
- Given an IP address, reputation filters will determine the reputation of the address.
- This is a more nuanced version of blacklists, as the reputation is on a scale, as opposed to a “yes or no” answer.
- Based on the reputation score, you can decide to reject, submit the message to more stringent tests, throttle messages from the IP, etc.
- Currently, most reputation services are commercial, and are tied to proprietary software or hardware.
- Many appliance vendors have their own reputation services.
- Some of these are:
  - Commercial: Habeas, Proofpoint Dynamic Reputation, SenderBase (IronPort), SenderScore (ReturnPath)
  - Open Source: KarmaSphere
- KarmaSphere is still early in its development, but holds promise.

I believe that reputation filtering shows a lot of promise, but currently its use is limited since most of the reputation technology out there is commercial and proprietary. Some systems like SenderBase will allow you to query their database to get a generic “good” or “bad” reputation for an IP address, but will not give you access to the true reputation score unless you license their technology. KarmaSphere is an open source system that could prove to be useful in the future, but I have not seen significant use of it yet.

Some URLs for these systems:

Habeas - http://www.habeas.com/
SenderBase - http://www.senderbase.org/
SenderScore - https://www.senderscore.org/

Karmasphere - http://www.karmasphere.com/



Tarpitting

- Tarpitting is another technique that relies on legitimate mail servers obeying the RFC standards.
- The idea is to delay responding to parts of the SMTP connection setup.
- The RFC defines that a mail server should wait at least 5 minutes for the initial SMTP connection and subsequent MAIL and RCPT commands.
- Spammers may also not wait for a connection if they do not get a response immediately.
- A legitimate mail server should wait for a proper SMTP response at each phase of the communication before continuing.
- Spammers will often blast out SMTP commands without waiting for a response.
- Tarpitting can be used to slow down or dissuade spammers from sending mail, or can be used to reject mail that does not follow the protocol.

This is another fairly recent technique that shows a lot of promise. Spammers rely on being able to send enormous quantities of mail in order to get a slim return rate, and by slowing down the amount they can send, tarpitting could cause it to become too costly for them to waste time sending mail. At the same time, delaying legitimate mail by a few minutes is not usually a big problem.

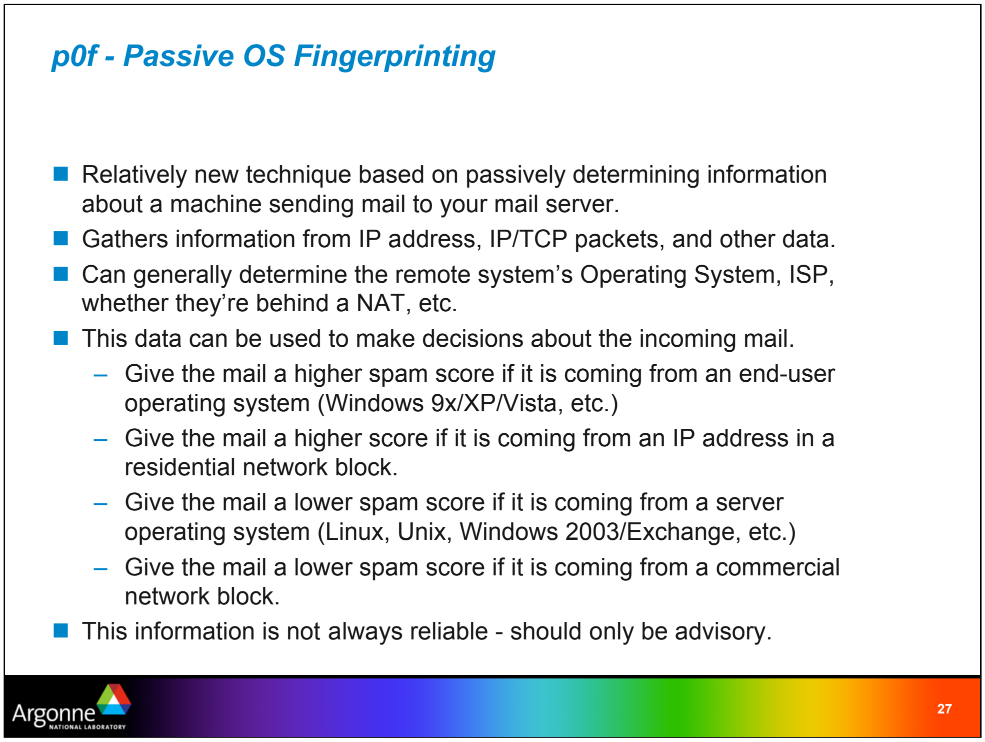
More information about tarpitting can be found here:

http://en.wikipedia.org/wiki/Tarpit_%28networking%29

There is a plugin for postgrey called targrey that adds tarpitting to the postgrey service.

http://k2net.hakuba.jp/targrey/index.en.html

To see an example of tarpitting in action, run the command ‘telnet openbsd.org 25’.



p0f - Passive OS Fingerprinting

- Relatively new technique based on passively determining information about a machine sending mail to your mail server.
- Gathers information from IP address, IP/TCP packets, and other data.
- Can generally determine the remote system's Operating System, ISP, whether they're behind a NAT, etc.
- This data can be used to make decisions about the incoming mail.
  - Give the mail a higher spam score if it is coming from an end-user operating system (Windows 9x/XP/Vista, etc.)
  - Give the mail a higher score if it is coming from an IP address in a residential network block.
  - Give the mail a lower spam score if it is coming from a server operating system (Linux, Unix, Windows 2003/Exchange, etc.)
  - Give the mail a lower spam score if it is coming from a commercial network block.
- This information is not always reliable - should only be advisory.

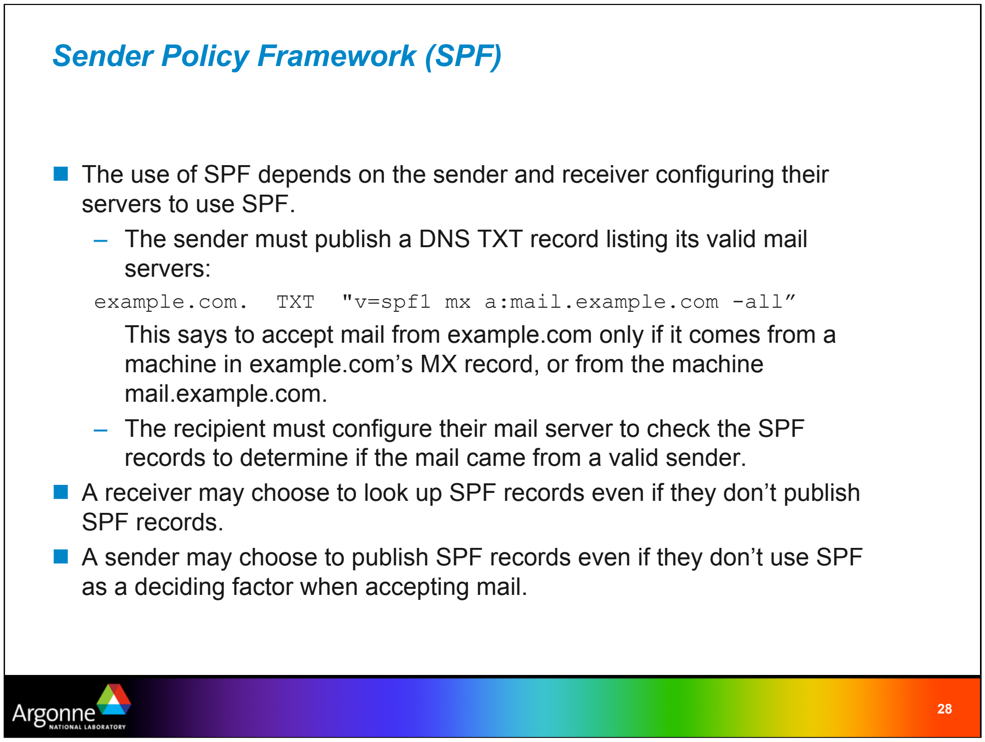
Argonne NATIONAL LABORATORY 27

This technique is fairly new, and I have not used it in production, but I think it holds a lot of promise. Most spam comes from commodity PCs running end-user OSes on home broadband networks, and by being a lot more suspicious of mail coming from those sources, you could stop a lot more spam. However, since many people do send legitimate mail from these types of computers, you do not want p0f to be used to make a final decision on what to do about an incoming message.

Here is the URL for the p0f homepage:

http://lcamtuf.coredump.cx/p0f.shtml

A p0f plugin for postgrey is available at:
http://postgrey.schweikert.ch/patches/02-postgrey-p0f.patch



Sender Policy Framework (SPF)

- The use of SPF depends on the sender and receiver configuring their servers to use SPF.
  - The sender must publish a DNS TXT record listing its valid mail servers:

```
example.com.  TXT  "v=spf1 mx a:mail.example.com -all”
```

  This says to accept mail from example.com only if it comes from a machine in example.com’s MX record, or from the machine mail.example.com.
  - The recipient must configure their mail server to check the SPF records to determine if the mail came from a valid sender.
- A receiver may choose to look up SPF records even if they don’t publish SPF records.
- A sender may choose to publish SPF records even if they don’t use SPF as a deciding factor when accepting mail.

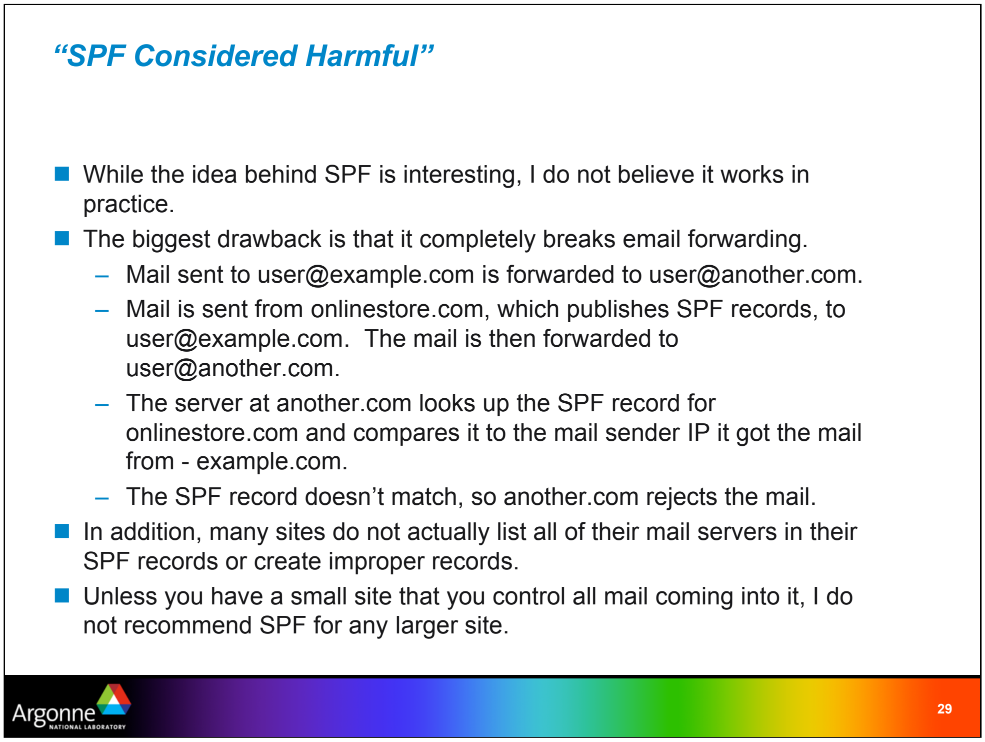
This is the technique that seems to inflame passions more than any other anti-spam technique. My personal feeling is that I do not believe that SPF is an effective tool for fighting spam - but more on that on the next page.

Here are some URLs regarding SPF:

Sender Policy Framework - http://en.wikipedia.org/wiki/Sender_Policy_Framework

OpenSPF - http://www.openspf.org/

SPF RFC - http://tools.ietf.org/html/rfc4408

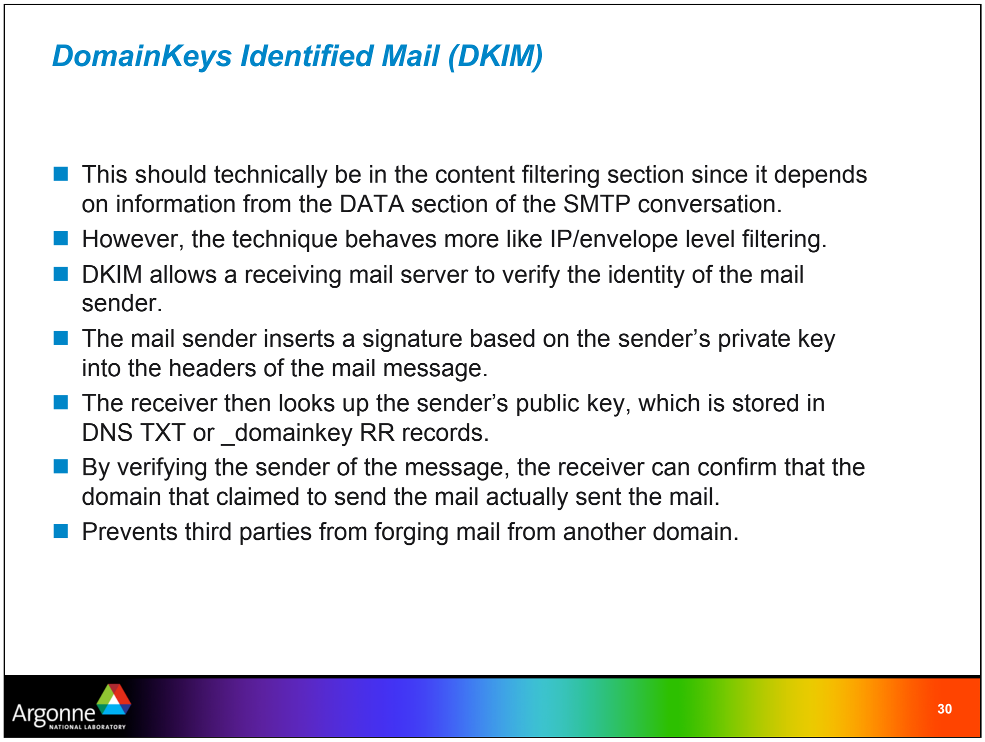


"SPF Considered Harmful"

- While the idea behind SPF is interesting, I do not believe it works in practice.
- The biggest drawback is that it completely breaks email forwarding.
  - Mail sent to user@example.com is forwarded to user@another.com.
  - Mail is sent from onlinestore.com, which publishes SPF records, to user@example.com. The mail is then forwarded to user@another.com.
  - The server at another.com looks up the SPF record for onlinestore.com and compares it to the mail sender IP it got the mail from - example.com.
  - The SPF record doesn't match, so another.com rejects the mail.
- In addition, many sites do not actually list all of their mail servers in their SPF records or create improper records.
- Unless you have a small site that you control all mail coming into it, I do not recommend SPF for any larger site.

When I tried to implement SPF at my site, the mail forwarding issue came up within an hour. In addition, I found that a certain large, unnamed mail provider listed SPF records and then sent mail from hosts not in those records. Ironically, they are also one of the biggest sites pushing SPF. After a few hours of fielding calls, I determined that SPF was doing more harm than good, and disabled it. Also, the DNS TXT records that SPF uses are not intended for storing that kind of data, and may conflict with other software.

The title of this slide came from this article:
http://bradknowles.typepad.com./considered_harmful/2004/05/spf.html



DomainKeys Identified Mail (DKIM)

- This should technically be in the content filtering section since it depends on information from the DATA section of the SMTP conversation.
- However, the technique behaves more like IP/envelope level filtering.
- DKIM allows a receiving mail server to verify the identity of the mail sender.
- The mail sender inserts a signature based on the sender’s private key into the headers of the mail message.
- The receiver then looks up the sender’s public key, which is stored in DNS TXT or _domainkey RR records.
- By verifying the sender of the message, the receiver can confirm that the domain that claimed to send the mail actually sent the mail.
- Prevents third parties from forging mail from another domain.

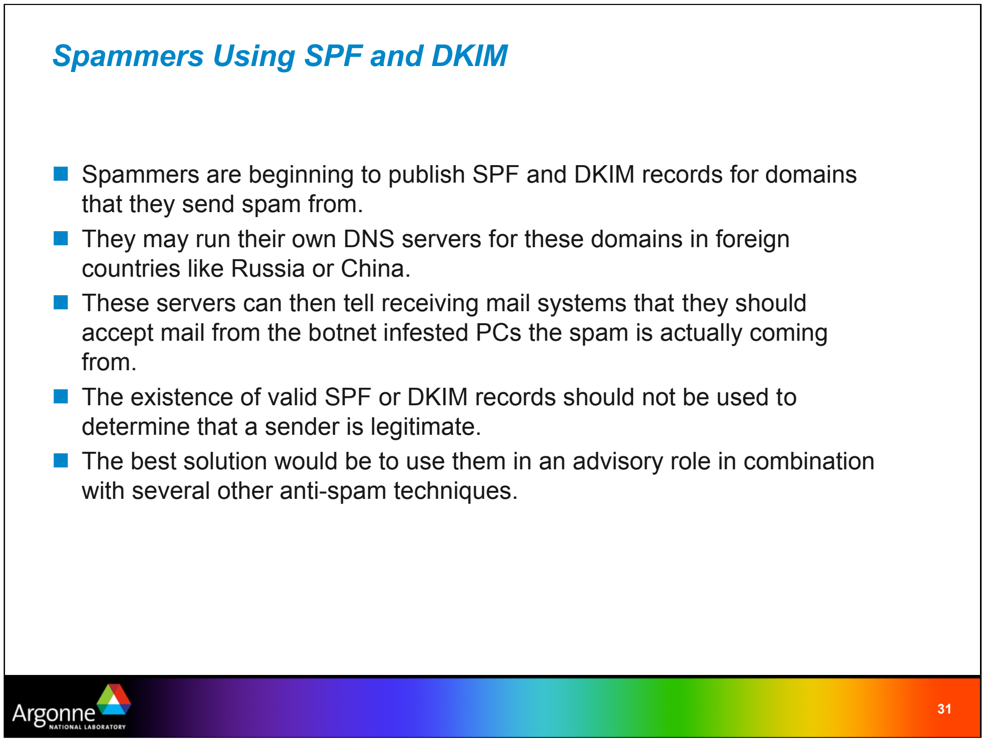
This is another technique that has vocal supporters and detractors. I think it could be a useful technique to try to fight phishing attacks by having banks use it to verify that they were the actual sender of some sort of financial email, but at the same time, without universal adoption of this sort of technology, phishing attacks will always continue. This also is a technique that uses DNS in a way that it was not originally intended for. There have been a lot of similar techniques like DKIM that have been proposed (Sender ID, etc.), but I do not believe that any of them will gain widespread acceptance as long as we rely on SMTP - it’s simply not designed to be secure, and all of these techniques are essentially just patches.

For more information on DKIM:

http://www.dkim.org/

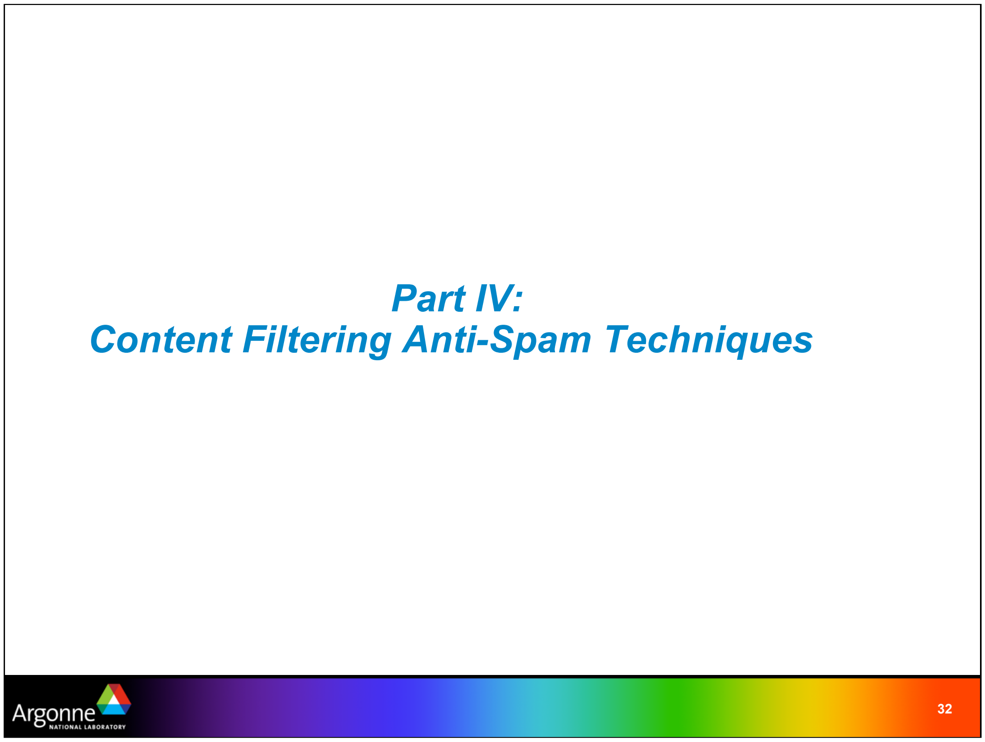
http://en.wikipedia.org/wiki/DomainKeys_Identified_Mail

RFCs - http://tools.ietf.org/html/rfc4870, http://tools.ietf.org/html/rfc4871

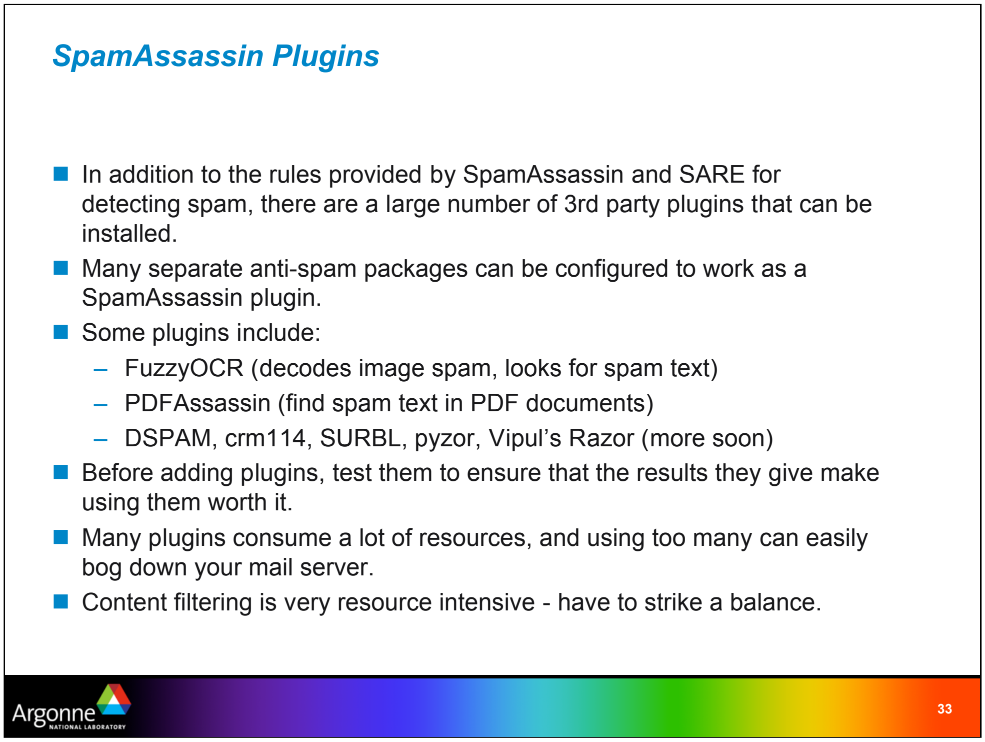


Spammers Using SPF and DKIM

- Spammers are beginning to publish SPF and DKIM records for domains that they send spam from.
- They may run their own DNS servers for these domains in foreign countries like Russia or China.
- These servers can then tell receiving mail systems that they should accept mail from the botnet infested PCs the spam is actually coming from.
- The existence of valid SPF or DKIM records should not be used to determine that a sender is legitimate.
- The best solution would be to use them in an advisory role in combination with several other anti-spam techniques.



Part IV: Content Filtering Anti-Spam Techniques



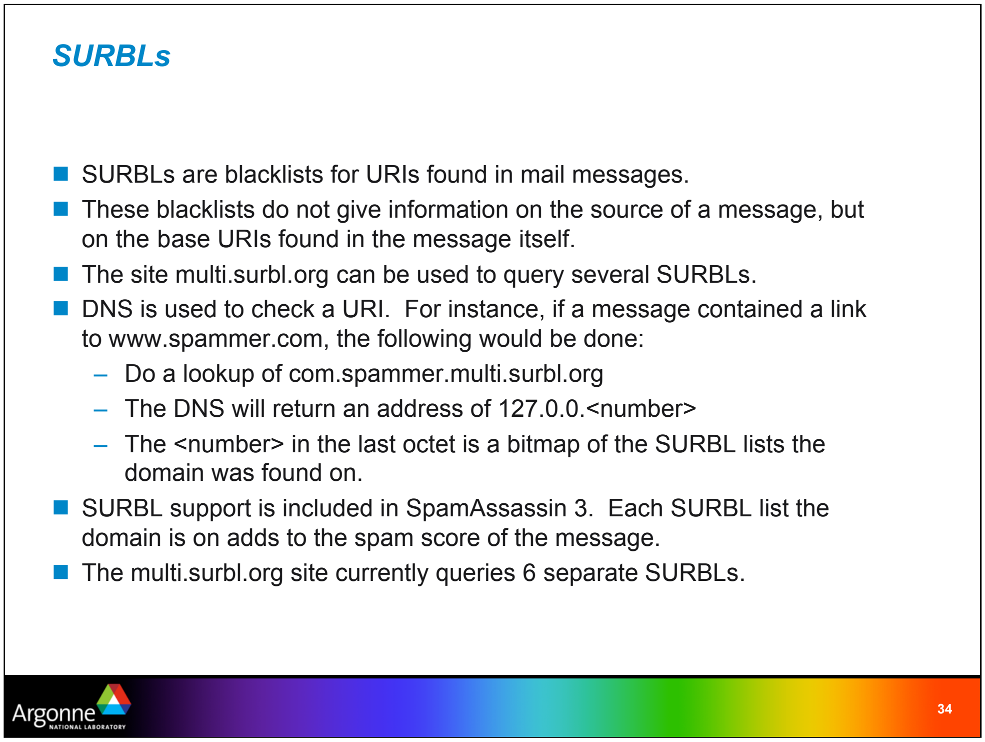
SpamAssassin Plugins

- In addition to the rules provided by SpamAssassin and SARE for detecting spam, there are a large number of 3rd party plugins that can be installed.
- Many separate anti-spam packages can be configured to work as a SpamAssassin plugin.
- Some plugins include:
  - FuzzyOCR (decodes image spam, looks for spam text)
  - PDFAssassin (find spam text in PDF documents)
  - DSPAM, crm114, SURBL, pyzor, Vipul’s Razor (more soon)
- Before adding plugins, test them to ensure that the results they give make using them worth it.
- Many plugins consume a lot of resources, and using too many can easily bog down your mail server.
- Content filtering is very resource intensive - have to strike a balance.

For more information on SpamAssassin plugins and the software mentioned here, here are some URLs:

http://wiki.apache.org/spamassassin/CustomPlugins

Fuzzy OCR - http://fuzzyocr.own-hero.net/



SURBLs

- SURBLs are blacklists for URIs found in mail messages.
- These blacklists do not give information on the source of a message, but on the base URIs found in the message itself.
- The site multi.surbl.org can be used to query several SURBLs.
- DNS is used to check a URI. For instance, if a message contained a link to www.spammer.com, the following would be done:
  - Do a lookup of com.spammer.multi.surbl.org
  - The DNS will return an address of 127.0.0.<number>
  - The <number> in the last octet is a bitmap of the SURBL lists the domain was found on.
- SURBL support is included in SpamAssassin 3. Each SURBL list the domain is on adds to the spam score of the message.
- The multi.surbl.org site currently queries 6 separate SURBLs.

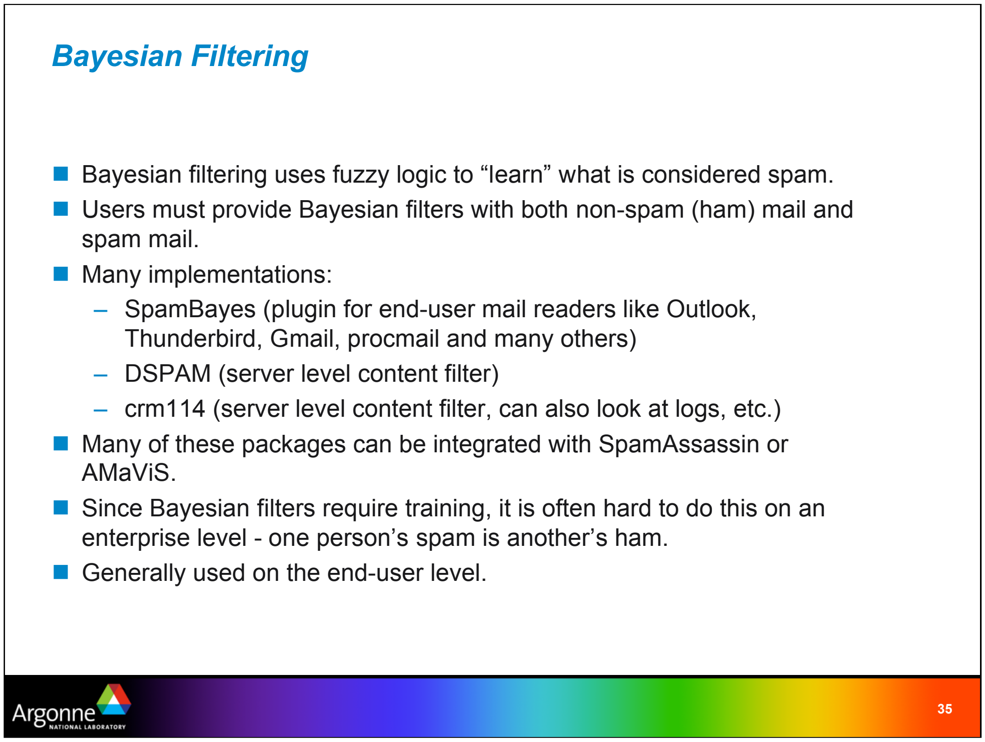
For more information on SURBL, and data on the specific lists that are use, visit:

http://www.surbl.org/

http://www.uribl.com/

To check if a domain is listed on a SURBL list, SARE has a web-based SURBL checker at the following URL:

http://www.rulesemporium.com/cgi-bin/uribl.cgi



Bayesian Filtering

- Bayesian filtering uses fuzzy logic to “learn” what is considered spam.
- Users must provide Bayesian filters with both non-spam (ham) mail and spam mail.
- Many implementations:
  - SpamBayes (plugin for end-user mail readers like Outlook, Thunderbird, Gmail, procmail and many others)
  - DSPAM (server level content filter)
  - crm114 (server level content filter, can also look at logs, etc.)
- Many of these packages can be integrated with SpamAssassin or AMaViS.
- Since Bayesian filters require training, it is often hard to do this on an enterprise level - one person’s spam is another’s ham.
- Generally used on the end-user level.

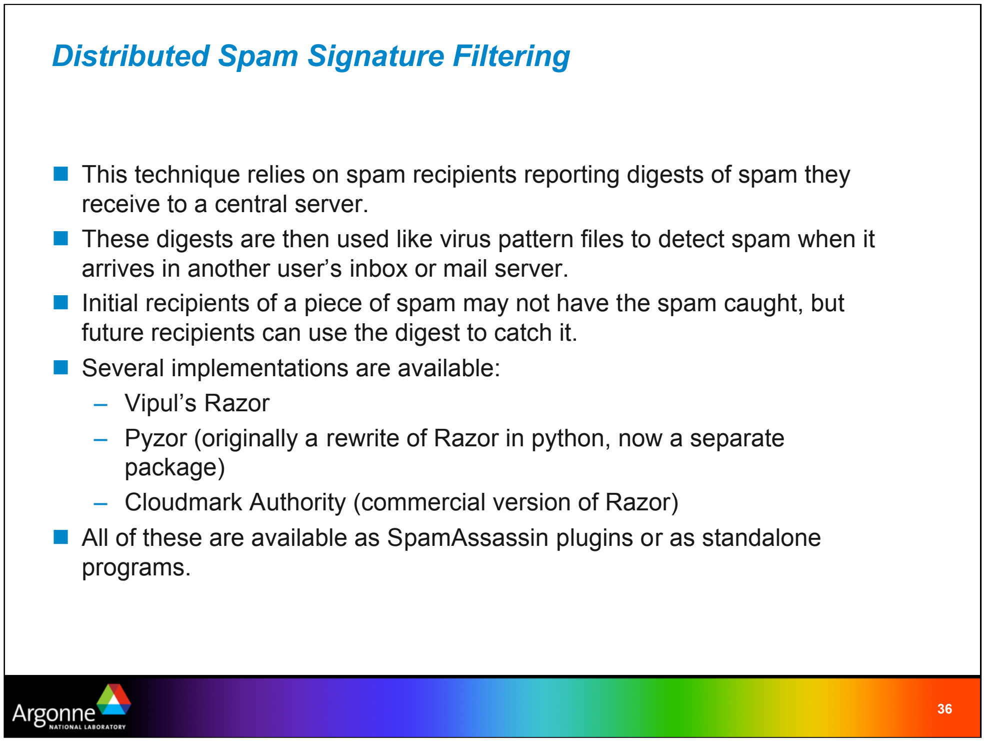
Bayesian filtering is based on Bayes’ Theorem, which, in the context of spam, “says that the probability that an email is spam, given that it has certain words in it, is equal to the probability of finding those certain words in spam email, times the probability that any email is spam, divided by the probability of finding those words in any email”. [cite: Wikipedia article on Bayesian filtering, URL given below.]

For more information on Bayesian filtering:

http://en.wikipedia.org/wiki/Bayesian_spam_filtering

Here are the URLs for the software packages mentioned above:

SpamBayes - http://spambayes.sourceforge.net/
DSpam - http://dspam.nuclearelephant.com/
crm114 - http://crm114.sourceforge.net/



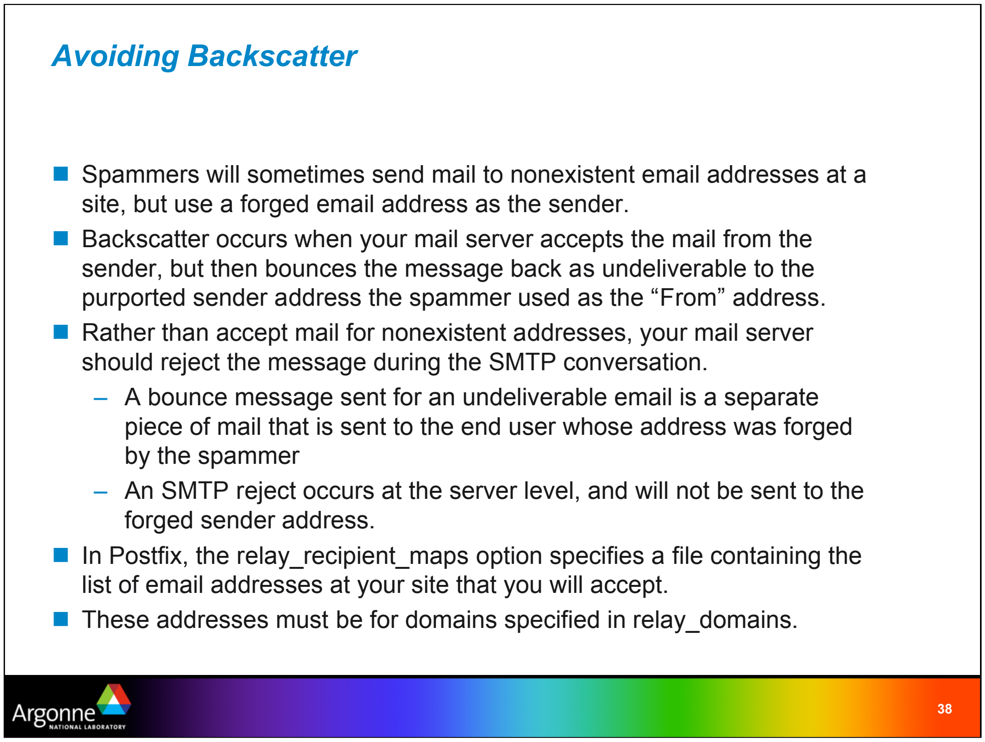
Distributed Spam Signature Filtering

- This technique relies on spam recipients reporting digests of spam they receive to a central server.
- These digests are then used like virus pattern files to detect spam when it arrives in another user’s inbox or mail server.
- Initial recipients of a piece of spam may not have the spam caught, but future recipients can use the digest to catch it.
- Several implementations are available:
  - Vipul’s Razor
  - Pyzor (originally a rewrite of Razor in python, now a separate package)
  - Cloudmark Authority (commercial version of Razor)
- All of these are available as SpamAssassin plugins or as standalone programs.

The URLs for the software packages listed above are:

Vipul's Razor - http://razor.sourceforge.net/
Pyzor - http://pyzor.sourceforge.net/
Cloudmark Authority - http://www.cloudmark.com/

Part V:
Other Anti-Spam Techniques



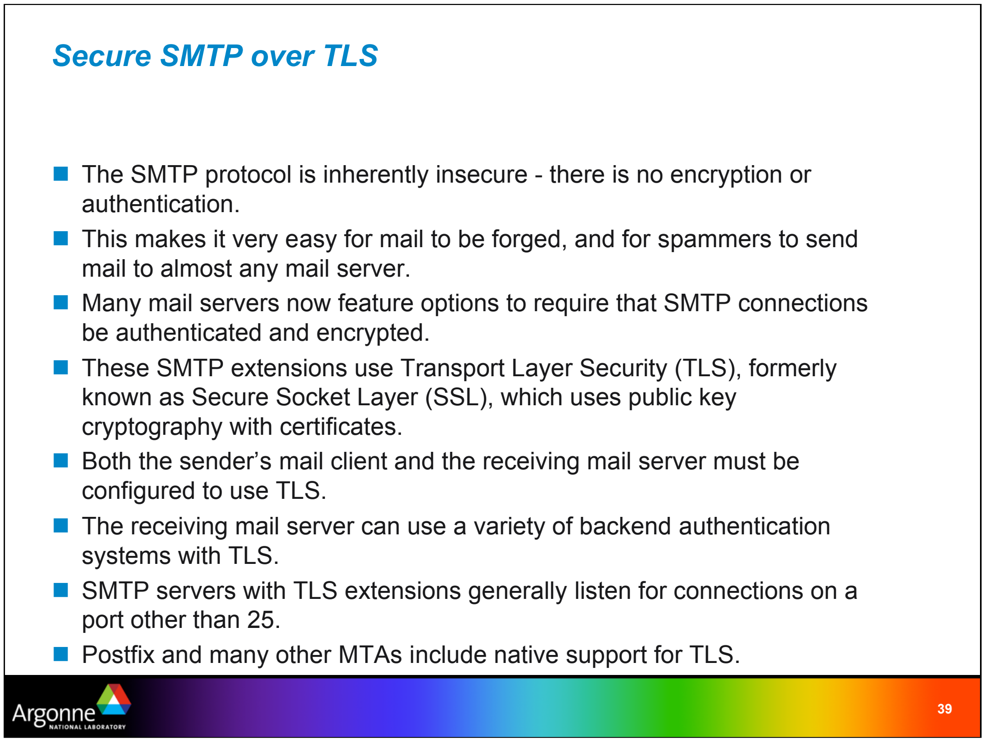
Avoiding Backscatter

- Spammers will sometimes send mail to nonexistent email addresses at a site, but use a forged email address as the sender.
- Backscatter occurs when your mail server accepts the mail from the sender, but then bounces the message back as undeliverable to the purported sender address the spammer used as the “From” address.
- Rather than accept mail for nonexistent addresses, your mail server should reject the message during the SMTP conversation.
  - A bounce message sent for an undeliverable email is a separate piece of mail that is sent to the end user whose address was forged by the spammer
  - An SMTP reject occurs at the server level, and will not be sent to the forged sender address.
- In Postfix, the relay_recipient_maps option specifies a file containing the list of email addresses at your site that you will accept.
- These addresses must be for domains specified in relay_domains.

More information on backscatter can be found here:

http://en.wikipedia.org/wiki/Backscatter#Backscatter_of_email_spam

If you run a mail gateway for many internal mail servers, the best way to avoid backscatter is to have the list of all users on the internal mail servers put in the relay_recipient_maps on the gateway. This allows you to reject mail to unknown addresses at the gateway level, before it ever reaches the final destination.



Secure SMTP over TLS

- The SMTP protocol is inherently insecure - there is no encryption or authentication.
- This makes it very easy for mail to be forged, and for spammers to send mail to almost any mail server.
- Many mail servers now feature options to require that SMTP connections be authenticated and encrypted.
- These SMTP extensions use Transport Layer Security (TLS), formerly known as Secure Socket Layer (SSL), which uses public key cryptography with certificates.
- Both the sender's mail client and the receiving mail server must be configured to use TLS.
- The receiving mail server can use a variety of backend authentication systems with TLS.
- SMTP servers with TLS extensions generally listen for connections on a port other than 25.
- Postfix and many other MTAs include native support for TLS.

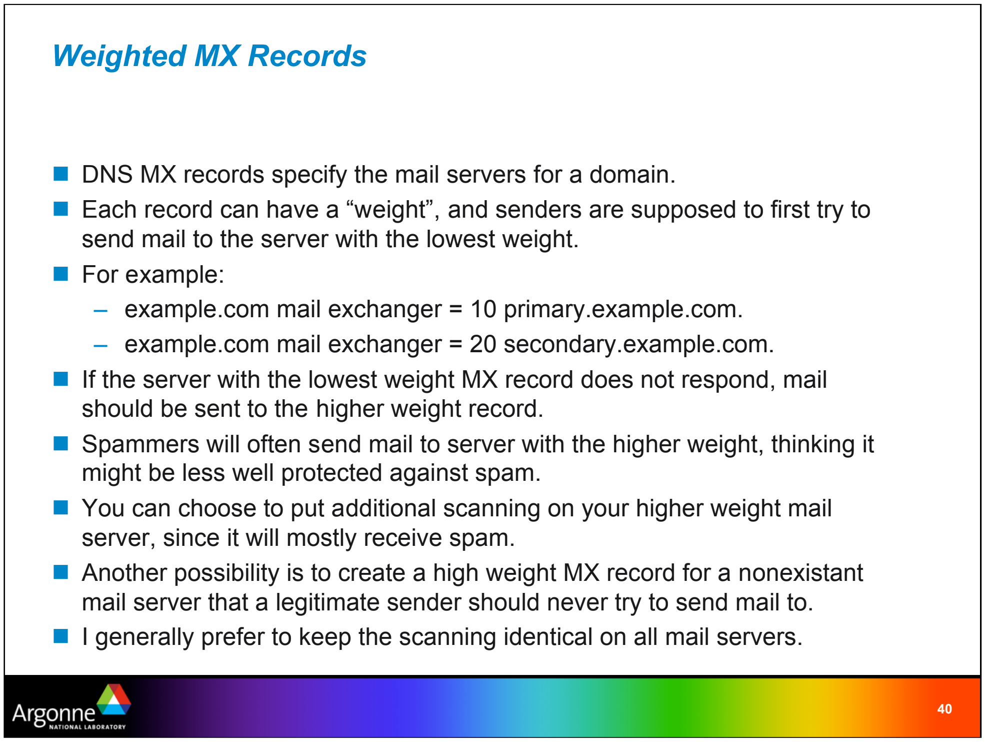
Configuring SMTP with TLS can be fairly complicated. The following web pages provide much greater detail:

http://en.wikipedia.org/wiki/Extended_SMTP
http://en.wikipedia.org/wiki/Transport_layer_security

RFC 3207, "Secure SMTP over Transport Layer Security" - http://tools.ietf.org/html/rfc3207

Information on the Postfix implementation of TLS can be found here:

http://www.postfix.org/TLS_README.html

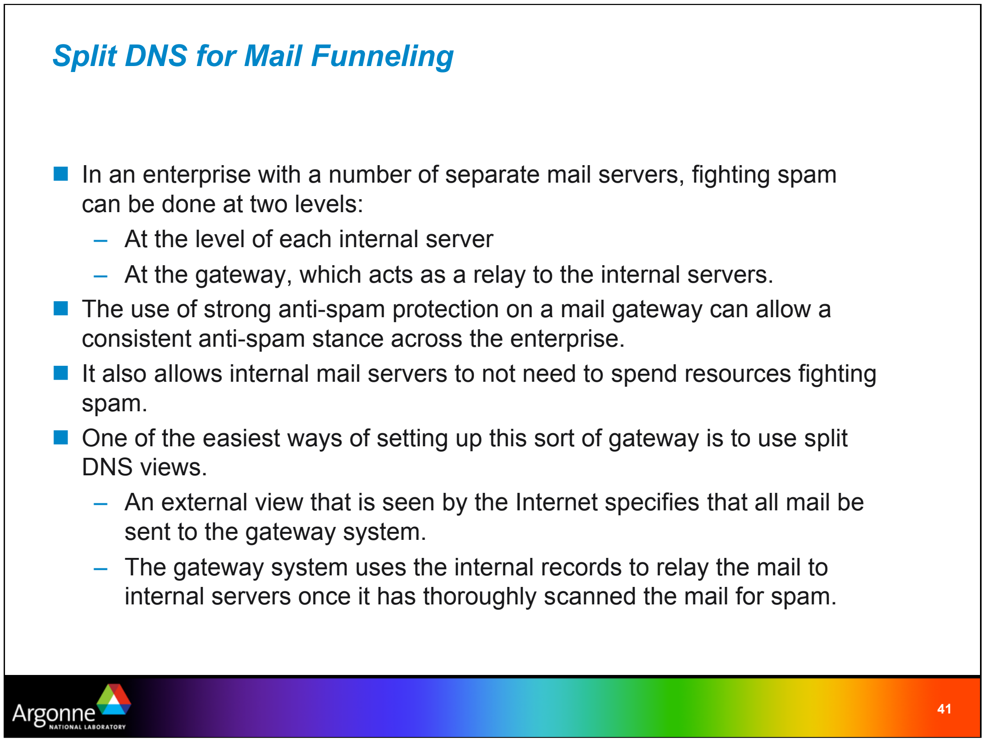


Weighted MX Records

- DNS MX records specify the mail servers for a domain.
- Each record can have a “weight”, and senders are supposed to first try to send mail to the server with the lowest weight.
- For example:
  - example.com mail exchanger = 10 primary.example.com.
  - example.com mail exchanger = 20 secondary.example.com.
- If the server with the lowest weight MX record does not respond, mail should be sent to the higher weight record.
- Spammers will often send mail to server with the higher weight, thinking it might be less well protected against spam.
- You can choose to put additional scanning on your higher weight mail server, since it will mostly receive spam.
- Another possibility is to create a high weight MX record for a nonexistant mail server that a legitimate sender should never try to send mail to.
- I generally prefer to keep the scanning identical on all mail servers.

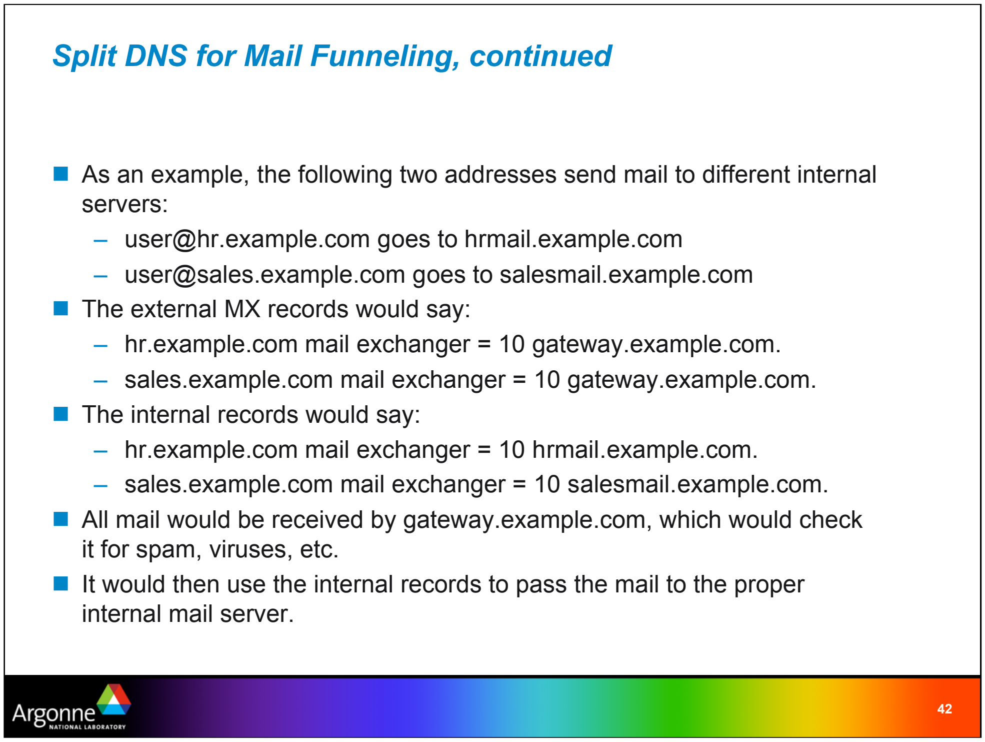
I have heard spirited debate on the idea of creating nonexistent high weight MX records. Some people believe that it works well, but others have denounced it as another abuse of DNS. I am not convinced that it would add that much to an anti-spam system. There are going to be times when legitimate senders try to send mail to the server and fail, and spammers will likely just try one of the lower weight MX records if they can’t connect to a nonexistent high weight record.

That said, I can see some truth to the argument, as my system logs indicate that on my primary (low weight) MX host, 80-85% of the incoming mail is spam, while the percentage on my backup (high weight) MX host is 95-99% spam. But on the flip side, the amount of mail sent to my backup MX host is much less than the amount sent to my primary MX.



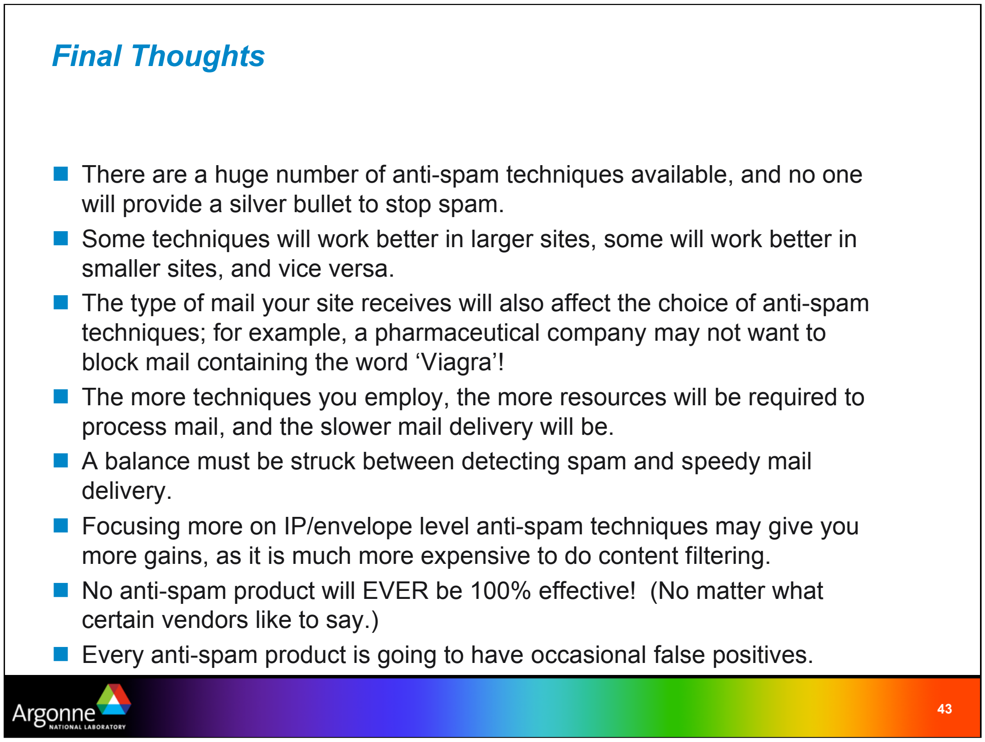
Split DNS for Mail Funneling

- In an enterprise with a number of separate mail servers, fighting spam can be done at two levels:
  - At the level of each internal server
  - At the gateway, which acts as a relay to the internal servers.
- The use of strong anti-spam protection on a mail gateway can allow a consistent anti-spam stance across the enterprise.
- It also allows internal mail servers to not need to spend resources fighting spam.
- One of the easiest ways of setting up this sort of gateway is to use split DNS views.
  - An external view that is seen by the Internet specifies that all mail be sent to the gateway system.
  - The gateway system uses the internal records to relay the mail to internal servers once it has thoroughly scanned the mail for spam.



Split DNS for Mail Funneling, continued

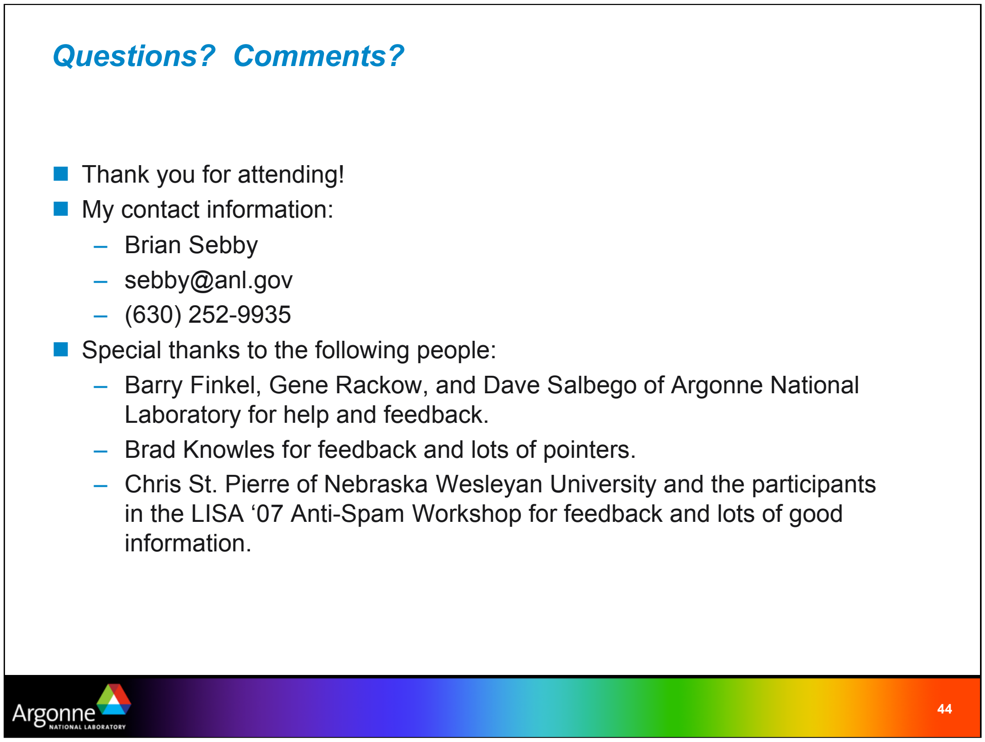
- As an example, the following two addresses send mail to different internal servers:
  - user@hr.example.com goes to hrmail.example.com
  - user@sales.example.com goes to salesmail.example.com
- The external MX records would say:
  - hr.example.com mail exchanger = 10 gateway.example.com.
  - sales.example.com mail exchanger = 10 gateway.example.com.
- The internal records would say:
  - hr.example.com mail exchanger = 10 hrmail.example.com.
  - sales.example.com mail exchanger = 10 salesmail.example.com.
- All mail would be received by gateway.example.com, which would check it for spam, viruses, etc.
- It would then use the internal records to pass the mail to the proper internal mail server.



Final Thoughts

- There are a huge number of anti-spam techniques available, and no one will provide a silver bullet to stop spam.
- Some techniques will work better in larger sites, some will work better in smaller sites, and vice versa.
- The type of mail your site receives will also affect the choice of anti-spam techniques; for example, a pharmaceutical company may not want to block mail containing the word ‘Viagra’!
- The more techniques you employ, the more resources will be required to process mail, and the slower mail delivery will be.
- A balance must be struck between detecting spam and speedy mail delivery.
- Focusing more on IP/envelope level anti-spam techniques may give you more gains, as it is much more expensive to do content filtering.
- No anti-spam product will EVER be 100% effective! (No matter what certain vendors like to say.)
- Every anti-spam product is going to have occasional false positives.

I once had a vendor leave a voice mail about their product that offered 100% detection with 0 false positives. I didn’t call them back.



Questions? Comments?

- Thank you for attending!
- My contact information:
  - Brian Sebby
  - sebby@anl.gov
  - (630) 252-9935
- Special thanks to the following people:
  - Barry Finkel, Gene Rackow, and Dave Salbego of Argonne National Laboratory for help and feedback.
  - Brad Knowles for feedback and lots of pointers.
  - Chris St. Pierre of Nebraska Wesleyan University and the participants in the LISA '07 Anti-Spam Workshop for feedback and lots of good information.

To read a summary of what was discussed at the LISA '07 Anti-Spam Workshop, look at page 108 in this PDF file:

http://www.usenix.org/publications/login/2008-02/openpdfs/LISA07reports.pdf