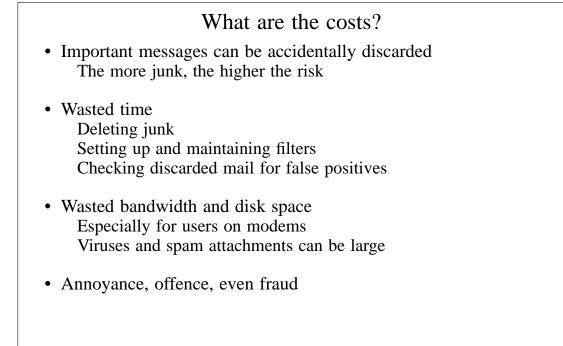
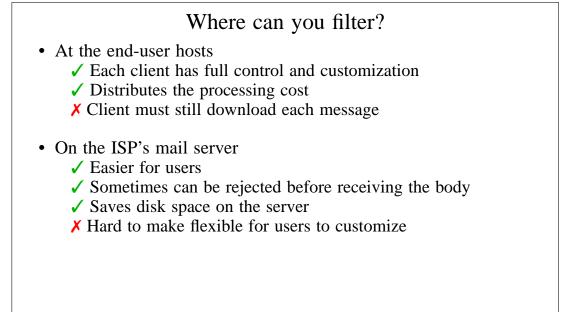


	What are the main sources of junk email?
• S	Spam Unsolicited, bulk email Often fraudulent – penis enlargement, lottery scams, close relatives of African presidents, etc. Low response rate => high volume sent
• \	Viruses, Trojan horses Infected machine sends out mails without the owner's knowledge
• N	Aalicious bounces These are called "collateral spam" or "Joe-jobs" Junk mail is sent with forged MAIL FROM Accepted by some intermediate MTA, but later it bounces Bounces go to innocent third party

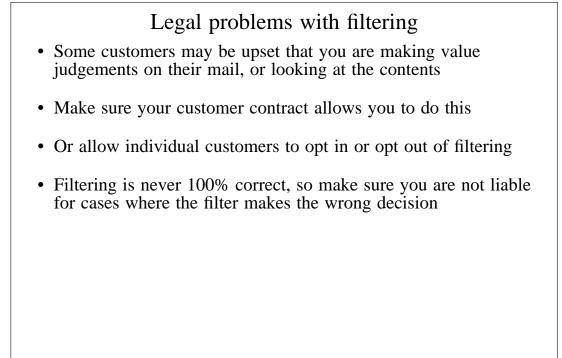


There are no easy answers!

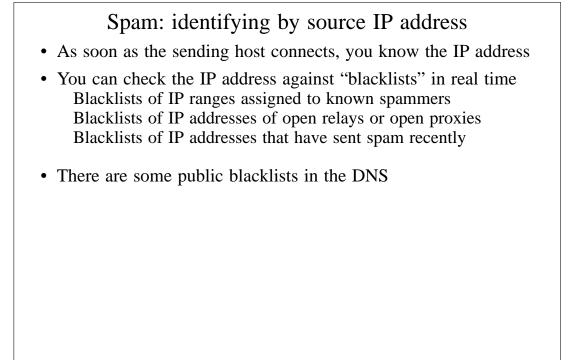


## The Joe-job problem

- Don't accept a message and then bounce it later If its sender is forged, we are creating a Joe-job
- Much better to reject at RCPT TO or DATA stages A real MTA sender will create a bounce Spamware will ignore the rejection
- For content filtering, we have to reject at DATA time If there are multiple recipients, that rejects it for all This makes individual opt-in/opt-out difficult
- What about accepting and just discarding junk? Risky because of false positives If a real message is rejected by mistake, nobody knows



#### Viruses in email • The volume of virus mail is now huge It is amazing how innocent some users are Cambridge University rejects 80% of the email it is offered See http://canvas.csi.cam.ac.uk/stats/ppsw/index.html That excludes spam, which is tagged, not rejected • Like spam, current viruses have forged senders and headers • Naive implementation blocks all executable extensions Can block some legitimate messages Some viruses come in .zip files • The only sure test is to use a virus scanner Commercial solutions are expensive, may charge per-user Free solutions such as *clamav* are pretty good • New viruses are being written all the time Frequent updating of the signatures is important clamav can do this automatically



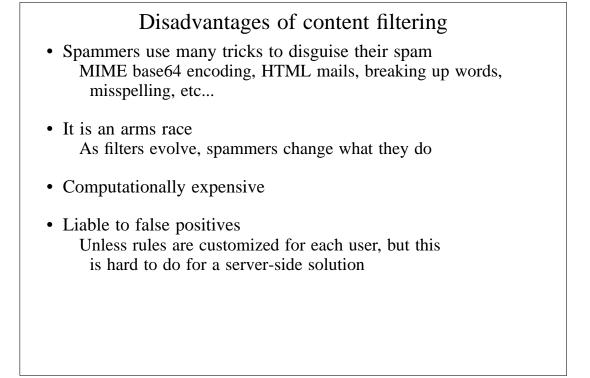
#### Using DNS black lists

- Advantages
  - ✓ Easy to configure
  - ✓ DNS lookups are relatively quick and cheap
  - ✓ Somebody else maintains the list
  - ✓ Mail is rejected before the body has been sent
- Disadvantages
  - X Will not catch all spam
  - X Not effective against viruses or collateral spam
  - X The lists come and go (legal threats from spammers)



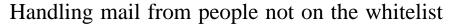
## Spam: identifying by content

- Spammers are sad and predictable
- A human can recognize spam very easily But it's harder to do it automatically
- Look for phrases that typically occur in spam
- Look for phrases type typically *do not* occur in spam This helps reduce false positives
- The ratio of the two indicates the likelihood of spam ... and how sure we are



#### Whitelists

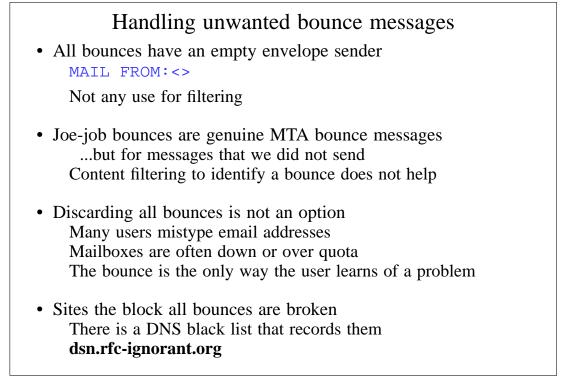
- Accept mail only from people we already know Effective at blocking spam and some viruses Start-up problem (see next slide)
- Actually, spammers could forge messages so that they appear to come from people we already know
- For now, they don't seem to be collecting information about who we associate with
- But viruses and Trojans often use local address books



- By password: e.g. a magic word in the **Subject:** header
- By content filter: e.g. a low spam score
- By challenge-reponse system Put mail in a hold queue and send back a message If the sender responds, they are whitelisted
- Challenge-response systems are not recommended
  - X Adds to the collateral spam problem
  - X Interacts badly with mailing lists
  - X Some people get very annoyed
  - X Difficult to deploy in a scalable way

## Disadvantages of whitelists

- Difficult/annoying for people to contact for the first time
- Difficult for a server-side solution Each user needs a separate list and a way to edit it Automatically whitelisting addresses we send TO isn't easy
- Filtering at the MAIL FROM stage is getting harder Envelope sender may differ from **From:** in headers It could even be different for every message someone sends
- Whitelists do not help with collateral spam (joe-jobs)

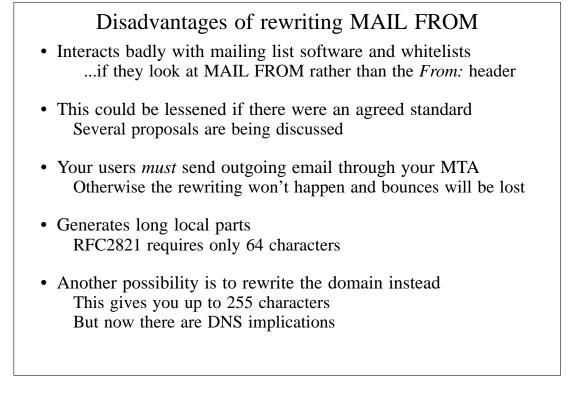


Associating bounces with messages we sent

- Bounce messages are not standardised in a way that allows this
- Only thing you can rely on is that bounces go to MAIL FROM
- One solution is to rewrite MAIL FROM

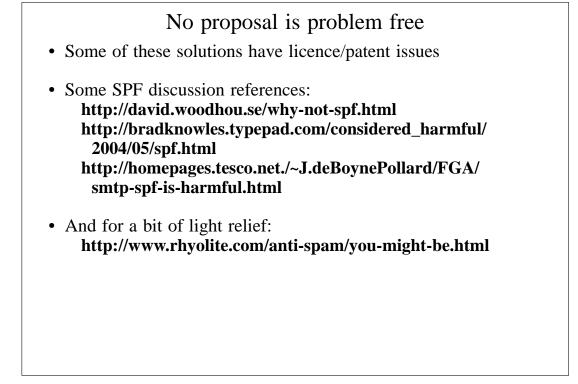
MAIL FROM:<user=ac7546dc@example.com>

- Change the magic value every day or so
- Check that incoming bounces quote a recent value
- If spammers collect the address, it is not valid for long
- Or use a cryptographic "cookie" (very hard to guess) Work is being done to refine these ideas
- This is not a spam solution; it's a Joe-job solution ...though it does kill spam sent with MAIL FROM:<>



# People are trying to find solutions BATV (Bounce Address Tag Validation)

- A scheme for adding tags to local parts
- CSA (Client SMTP Authorization) DNS lists which machines are permitted to send email
- SPF (Sender Policy Framework) Sender-ID is Microsoft's version of SPF DNS lists which hosts may use which envelope senders Completely breaks email forwarding Claims that it will kill all spam are exaggerations
- SRS (Sender Rewriting Scheme) An attempt to patch up SPF, but not problem-free
- Domainkeys (Yahoo!) or Identified Internet Mail (Cisco) Digitally sign messages with a per-domain private key The signature is placed in a header



Many options: what should you do?
• Use DNS blacklists Surprisingly effective Very easy to implement Low maintenance
• Consider implementing virus scanning and content filtering Opt-in users agree to let you do this Just tagging spam lets the user decide what to do
• Think about the resource costs These services are expensive to scale and manage Opt-in users pay extra?
• Advise users about client-side spam filters Bayesian filters and whitelists are more easily handled there Find those that work well with your client's software

