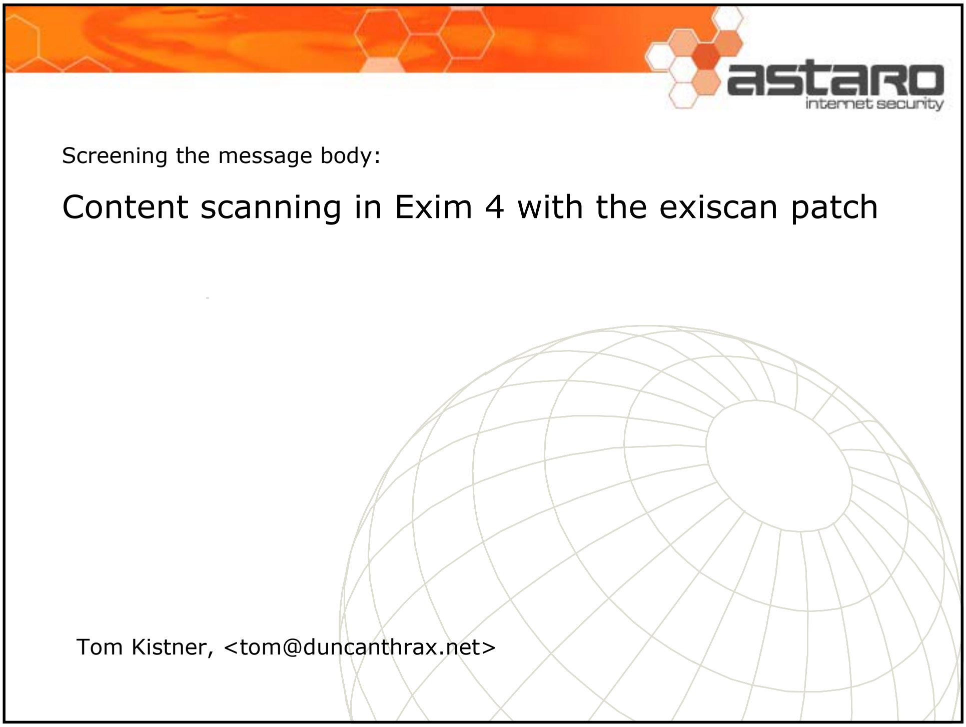
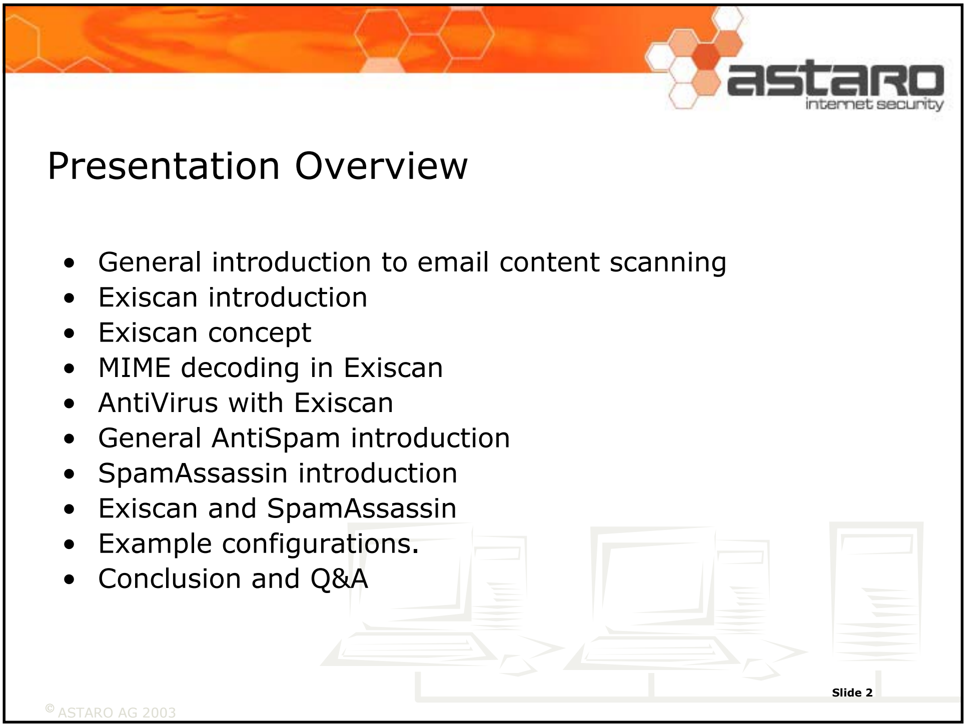


Screening the message body:

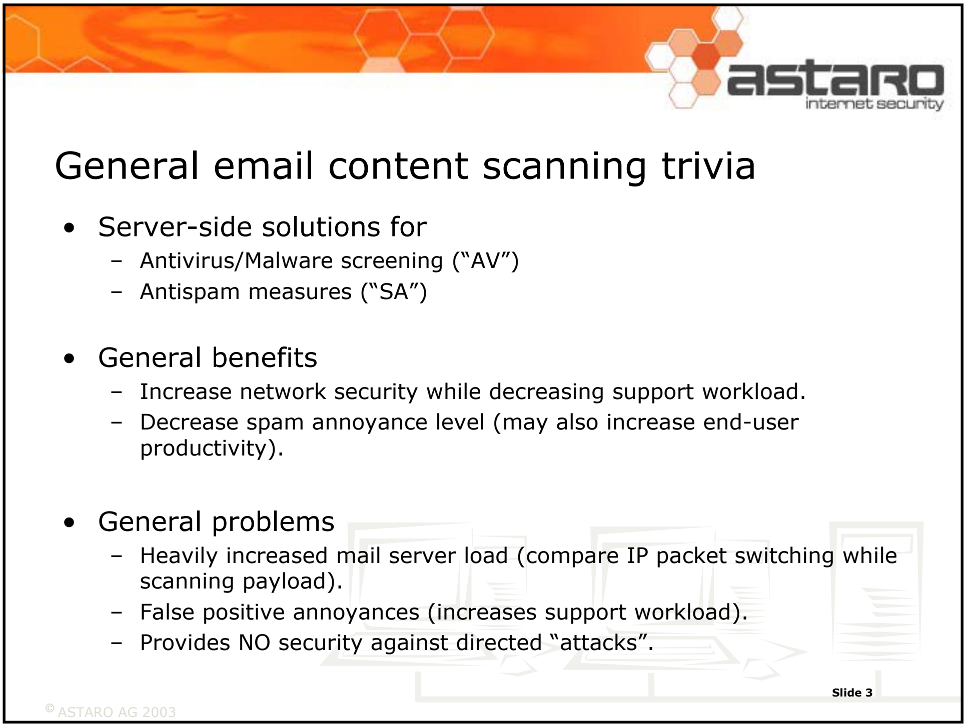
Content scanning in Exim 4 with the exiscan patch

Tom Kistner, <tom@duncanthrax.net>



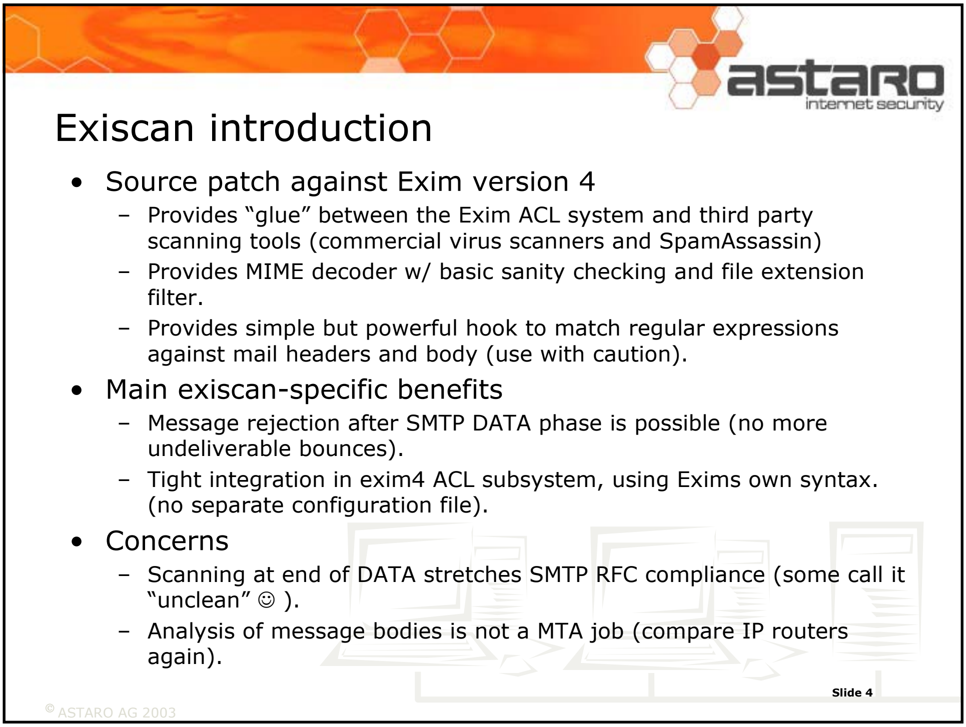
Presentation Overview

- General introduction to email content scanning
- Exiscan introduction
- Exiscan concept
- MIME decoding in Exiscan
- AntiVirus with Exiscan
- General AntiSpam introduction
- SpamAssassin introduction
- Exiscan and SpamAssassin
- Example configurations.
- Conclusion and Q&A



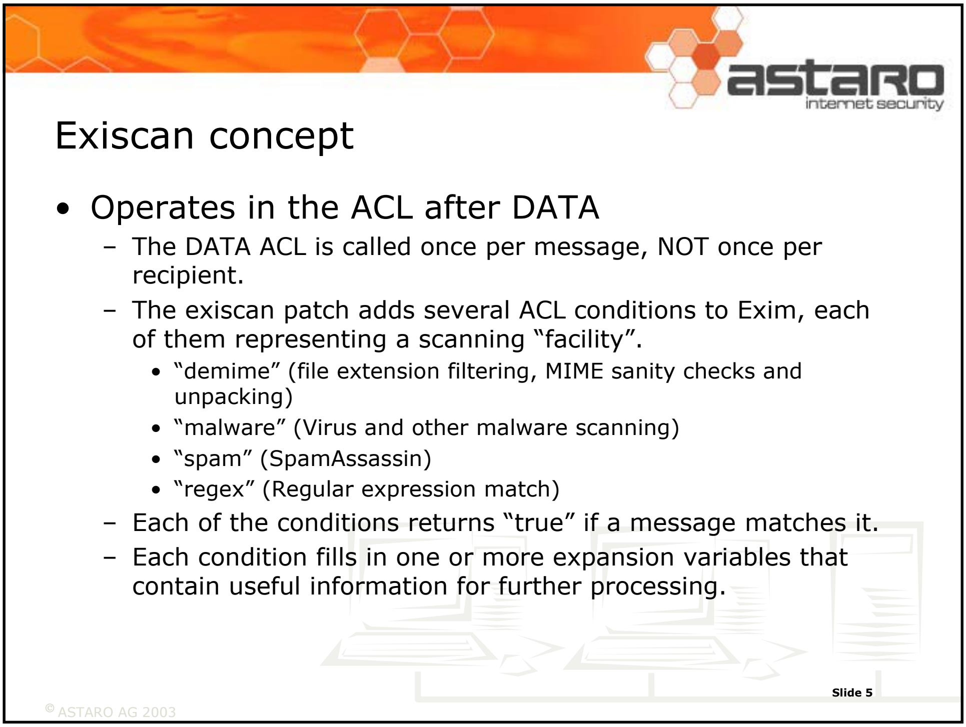
General email content scanning trivia

- Server-side solutions for
  - Antivirus/Malware screening (“AV”)
  - Antispam measures (“SA”)
- General benefits
  - Increase network security while decreasing support workload.
  - Decrease spam annoyance level (may also increase end-user productivity).
- General problems
  - Heavily increased mail server load (compare IP packet switching while scanning payload).
  - False positive annoyances (increases support workload).
  - Provides NO security against directed “attacks”.



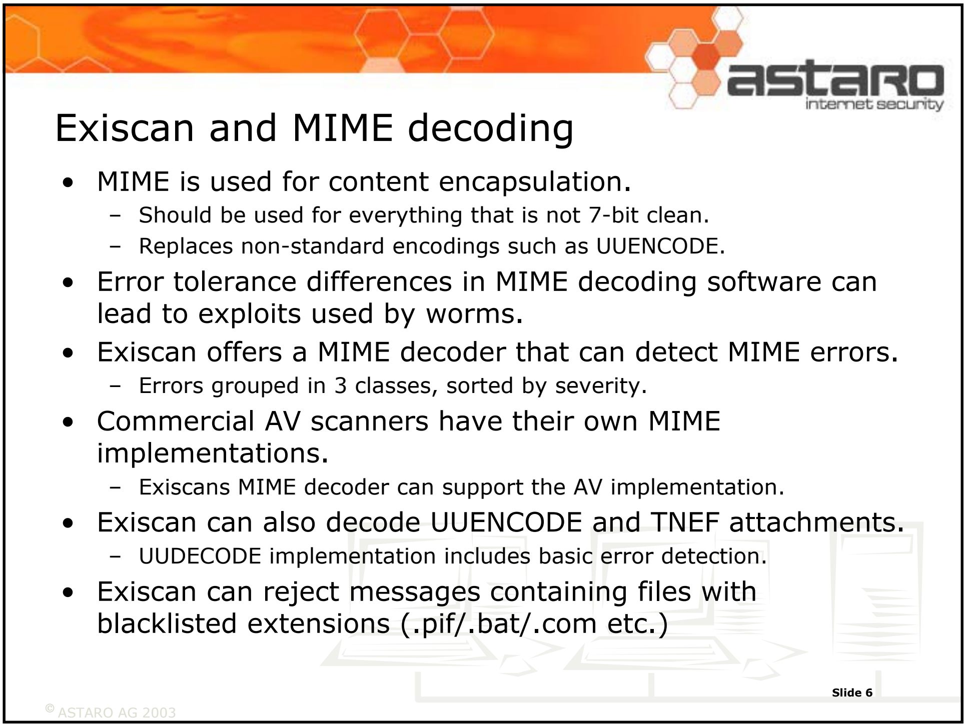
Exiscan introduction

- Source patch against Exim version 4
  - Provides “glue” between the Exim ACL system and third party scanning tools (commercial virus scanners and SpamAssassin)
  - Provides MIME decoder w/ basic sanity checking and file extension filter.
  - Provides simple but powerful hook to match regular expressions against mail headers and body (use with caution).
- Main exiscan-specific benefits
  - Message rejection after SMTP DATA phase is possible (no more undeliverable bounces).
  - Tight integration in exim4 ACL subsystem, using Exims own syntax. (no separate configuration file).
- Concerns
  - Scanning at end of DATA stretches SMTP RFC compliance (some call it “unclean” ☺).
  - Analysis of message bodies is not a MTA job (compare IP routers again).



Exiscan concept

- Operates in the ACL after DATA
  - The DATA ACL is called once per message, NOT once per recipient.
  - The exiscan patch adds several ACL conditions to Exim, each of them representing a scanning "facility".
    - "demime" (file extension filtering, MIME sanity checks and unpacking)
    - "malware" (Virus and other malware scanning)
    - "spam" (SpamAssassin)
    - "regex" (Regular expression match)
  - Each of the conditions returns "true" if a message matches it.
  - Each condition fills in one or more expansion variables that contain useful information for further processing.



Exiscan and MIME decoding

- MIME is used for content encapsulation.
  - Should be used for everything that is not 7-bit clean.
  - Replaces non-standard encodings such as UUENCODE.
- Error tolerance differences in MIME decoding software can lead to exploits used by worms.
- Exiscan offers a MIME decoder that can detect MIME errors.
  - Errors grouped in 3 classes, sorted by severity.
- Commercial AV scanners have their own MIME implementations.
  - Exiscans MIME decoder can support the AV implementation.
- Exiscan can also decode UUENCODE and TNEF attachments.
  - UUDECODE implementation includes basic error detection.
- Exiscan can reject messages containing files with blacklisted extensions (.pif/.bat/.com etc.)

Exiscan and AV

- Requires third party virus scanner
  - Generic support for scanners called via the shell (command line).
    - Slow, recommended only for low-volume systems.
  - Support for daemon-type scanners.
    - Fast operation, no forking or shell exec.
    - Supports Kapersky's "kavdaemon" and Sophos via the "Sophie" daemon.
    - Other daemon type scanners will be added over time.
- Typically very low false positive rate (next to none)
  - Recommended action is blackholing or rejection.
- Problems
  - Adds IO overhead (fast disk storage can help).
  - Scanner patterns must be kept up-to-date (consider automated update).

Advice on automated AV notifications

- Sender notifications
  "Your message contains a virus"
  - Useless with sender-faking worms (~95% of current malware traffic).
  - Creates confusion and adds to the problem.
  - Generates support calls on the sender side.
    - User: "I got a message that tells me I have a virus!"
    - Support: "Does it mention the term 'Klez'?"
- Recipient notifications
  "xyz@bongo.com was trying to send you a virus!"
  - Looks good from marketing point of view.
  - Useless, see above.
  - Generates support calls on your end (-> WORK!)
    - Users demand clarification and sometimes retaliation:
      "I knew that xyz hates me! Can't we send him some viruses back?"
- Postmaster notifications
  - Harmless, but not really necessary due to the low false positive rate.

Spam situation

- AntiSpam (AS) is the hype of 2002 and 2003
  - AV market is saturated. AS is a new opportunity for AV companies to increase slumping sales as worm flood is ebbing off.
  - Absolute spam message numbers increase as Spammer revenues go down due to increasing antispam measures -> those not deploying antispam software get flooded even more.
- Spam and AntiSpam collateral damage is huge.
  - High false positive rates (The "But I never got your email!" problem).
  - Forged headers cause massive complaint floods to innocent bystanders.
  - Email delivery reliability impaired by senseless "antispam measures".
- Possible measures.
  - Realtime Blackhole Lists (RBL), most of them host-based.
  - Filtering based on spam message characteristics.

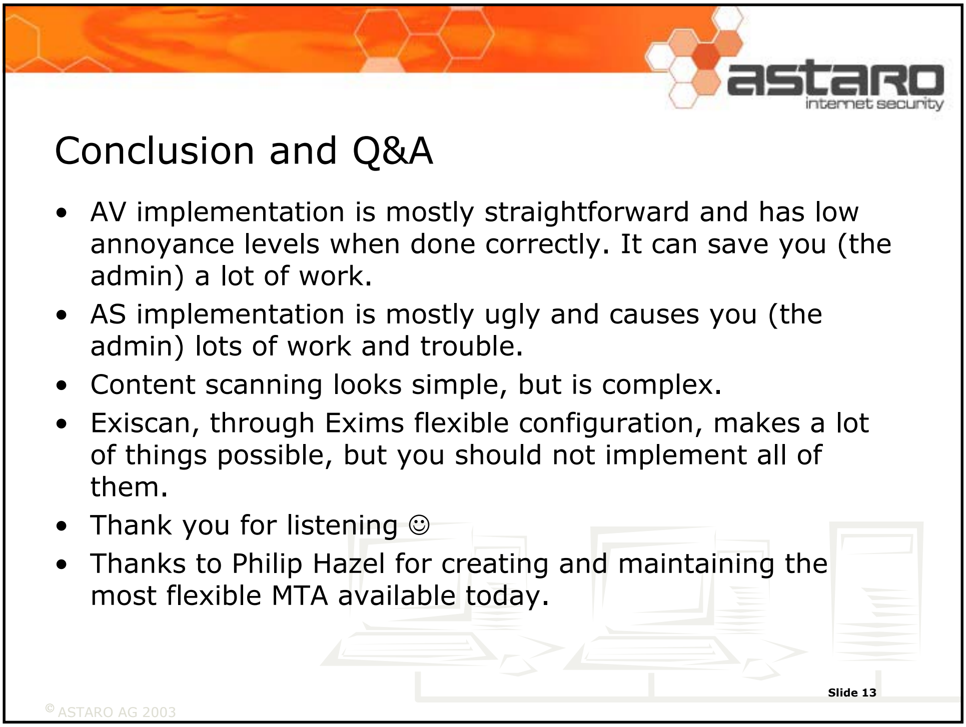
SpamAssassin overview

- SpamAssassin (SA) is a Spam detection engine written in Perl
  - Analyses message headers and body by running a large number of "tests".
  - Each successful test contributes a positive or negative value to a final "spam score".
  - Message is classified as spam if the score exceeds a "threshold" defined in the SA profile (default is "5").
  - SA can have multiple "profiles", affecting the threshold and weighting of individual tests.
  - SA has its own whitelist and blacklist system.
  - SA can query a number of non-local spam classification sources such as RBLs or Razor. Successful tests of those also contribute to the score.
  - SA can modify parts of the message to flag it as spam (These changes are NOT passed on by exiscan).
  - SA also features a self-learning bayesian component.

- Exiscan SA integration
  - Calls SA via the "spamd" daemon, passing a user (profile) name and the complete message.
  - Retrieves the spam score, the threshold and a human readable report.
  - Message modifications are made by Exim or the Exim System Filter, not SA.
- Problems
  - SA is very slow (CPU intensive), especially on larger messages.
  - False positive rate is fairly high.
  - SA is the most widely used AntiSpam tool, so Spammers work around its tests -> SA must be regularly updated to be effective.
- Performance Tips
  - Limit spam scanning to small message sizes (<80kB).
  - Build whitelist of trusted hosts that trade big mail volumes with your site, and extempt them from spam scanning.
  - Use exims RBL support to pre-filter known spammer hosts.

Example configurations

- There is no common "good" recipe for content filtering.
  - Implementation type depends on multiple factors. Examples:
    - Mail volume (higher volumes need more configuration tweaking).
    - Your policy enforcement style and end-user tolerance.
    - Company politics ("We need to add a 'guaranteed virus-free' footer!").
    - Legal issues when you have contracts with end users (ISPs).
- The Exiscan web site has an "Examples" document.
  - Provides some suggestions for commonly requested filtering tasks.
- Exiscan support is provided by the author and (increasingly) other users on the Exiscan mailing list.



Conclusion and Q&A

- AV implementation is mostly straightforward and has low annoyance levels when done correctly. It can save you (the admin) a lot of work.
- AS implementation is mostly ugly and causes you (the admin) lots of work and trouble.
- Content scanning looks simple, but is complex.
- Exiscan, through Exims flexible configuration, makes a lot of things possible, but you should not implement all of them.
- Thank you for listening ☺
- Thanks to Philip Hazel for creating and maintaining the most flexible MTA available today.