

---

# **SIEVE**

**Alexey Melnikov**

**Cyrus Daboo**

# What is SIEVE?

---

- **SIEVE is a Mail Filtering Language.**
- **Usually runs in SMTP (LMTP) servers or MUAs.**
- **Multiple implementations:**
  - Server side, like Exim, Cyrus/Isode, Sun, Sendmail.
  - User agents like Mulberry, Polymer, Ingo (web)
- **IETF SIEVE WG:**  
<http://www.ietf.org/html.charters/sieve-charter.html>

# What can SIEVE do right now (Sieve "tests")

---

- **Can examine**
  - header fields, addresses
  - SMTP envelope (if available)
  - message body (draft-ietf-sieve-body-02.txt)
  - spam scores and viruses (draft-ietf-sieve-spamtestbis-01.txt) – done externally
- **SIEVE now has support for variables (draft-ietf-sieve-variables-06.txt)**

# What can SIEVE do right now (Sieve "actions")

---

- **Sieve scripts can edit headers (draft-ietf-sieve-editheader-02.txt)**
- **Sieve scripts can set IMAP flags (draft-ietf-sieve-imapflags-01.txt)**
- **The received message can be**
  - filed into an IMAP mailbox
  - discarded/rejected
  - redirected to another address
  - send vacation replies (draft-ietf-sieve-vacation-03.txt)

# Application to Lemonade: Customizable notifications

---

- **draft-ietf-sieve-notify-00.txt:**

```
require ["notify", "fileinto", "variables"];
```

```
if header :contains "to" "sievemailinglist@example.org" {  
# :matches is used to get the value of the Subject header  
if header :matches "Subject" "*" {  
set "subject" "${1}";  
}  
}
```

```
# :matches is used to get the value of the From header  
if header :matches "From" "*" {  
set "from" "${1}";  
}
```

```
notify :low :message "[SIEVE] ${from}: ${subject}";  
fileinto "INBOX.sieve";  
}
```

# Application to Lemonade: Mailbox annotations

---

```
require ["notify", "mboxannotate", "variables", "envelope"];
```

```
if not annotation :private "INBOX" :matches  
"/vendor/isode/notification-uri"  
"none" {  
set "notif_uri" "${0}";
```

```
# :matches is used to get the MAIL FROM address  
if envelope :all :matches "from" "*" {  
set "env_from" " [really: ${1}]";  
}
```

```
# :matches is used to get the value of the Subject header  
if header :matches "Subject" "*" {  
set "subject" "${1}";  
}
```

```
# :matches is used to get the address from the From header  
if address :matches :all "from" "*" {  
set "from_addr" "${1}";  
}
```

```
notify :method "${notif_uri}"  
:message "${from_addr}${env_from}: ${subject}";  
}
```

# SIEVE script management/generation

---

- **SIEVE scripts can be managed using manage sieve protocol (draft-martin-managesieve-xx.txt).**
- **SIEVE generation can be hidden behind simple UI**
  - managing it from a small device becomes simple

# Future work

---

- **Creation of a SIEVE profile to work on IMAP flag changes/message deletion.**