# setroubleshoot

*A User Friendly Tool to Diagnose & Manage AVC Denials*

John Dennis - Red Hat

http://hosted.fedoraproject.org/projects/setroubleshoot

# Why is adoption slow?

- Developers won't work with SELinux enabled
- Perceived as creating new bugs
- Difficult to ascertain cause of software failure
- SELinux vendors can't support field problems
- System administrators are not trained to look for SELinux denials, cannot diagnose them, do not know how to resolve them, and ...

# ...The Release Notes Says:

We recommend disabling SELinux...

Our application does not work with SELinux...

If you experience any abnormal behavior you should disable SELinux...

# Error Reporting Misleading

- An SELinux denial reported as something else
  - Best case is EPERM, EACCESS: still misleading
- Traditionally look for DAC permission problem
  - But DAC is fine, HUH???
- SELinux denials appear in system log files
  - syslog
  - audit log

# Incomprehensible

- Even if error is correlated to SELinux the error message is incomprehensible

avc: denied { search } for comm="dbus-daemon" dev=hda5 egid=81 euid=81 exe="/bin/dbus-daemon" exit=-2 fsgid=81 fsuid=81 gid=81 items=0 name="yp" pid=2226 scontext=system_u:system_r:system_dbusd_t:s0 sgid=81 subj=system_u:system_r:system_dbusd_t:s0 suid=81 tclass=dir tcontext=system_u:object_r:var_yp_t:s0 tty=(none) uid=81

Should be:

SELinux prevented the dbus daemon from using NIS (yp)

# Denials Are Silent

- For practical purposes denials are silent
  - Only in log file
  - Little correlation to application
  - Incomprehensible
  - Little correlation to user action

# Goals

- Plug-in architecture for analysis modules
- Flexible alert mechanism
  - GUI pop up notification
  - Email notification
  - System monitoring integration
  - Not obnoxious

# Goals (continued)

- Easy review of alerts
- No dependencies outside of core Linux
- Both local & distributed monitoring
- Integration with bug reporting
- Query if alert represents known problem

# Implementation

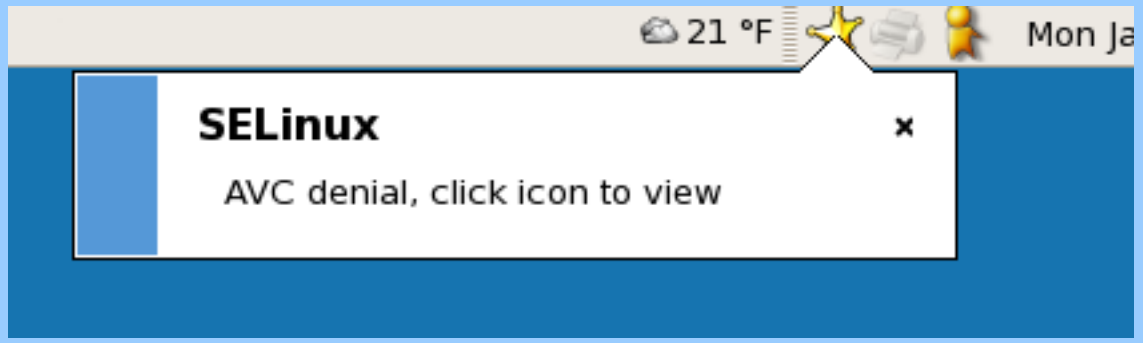
- Written in python
- XML storage and data exchange
- Client/Server model with RPC
- Highly asynchronous
  - Completely event driven model

# Show me the money!

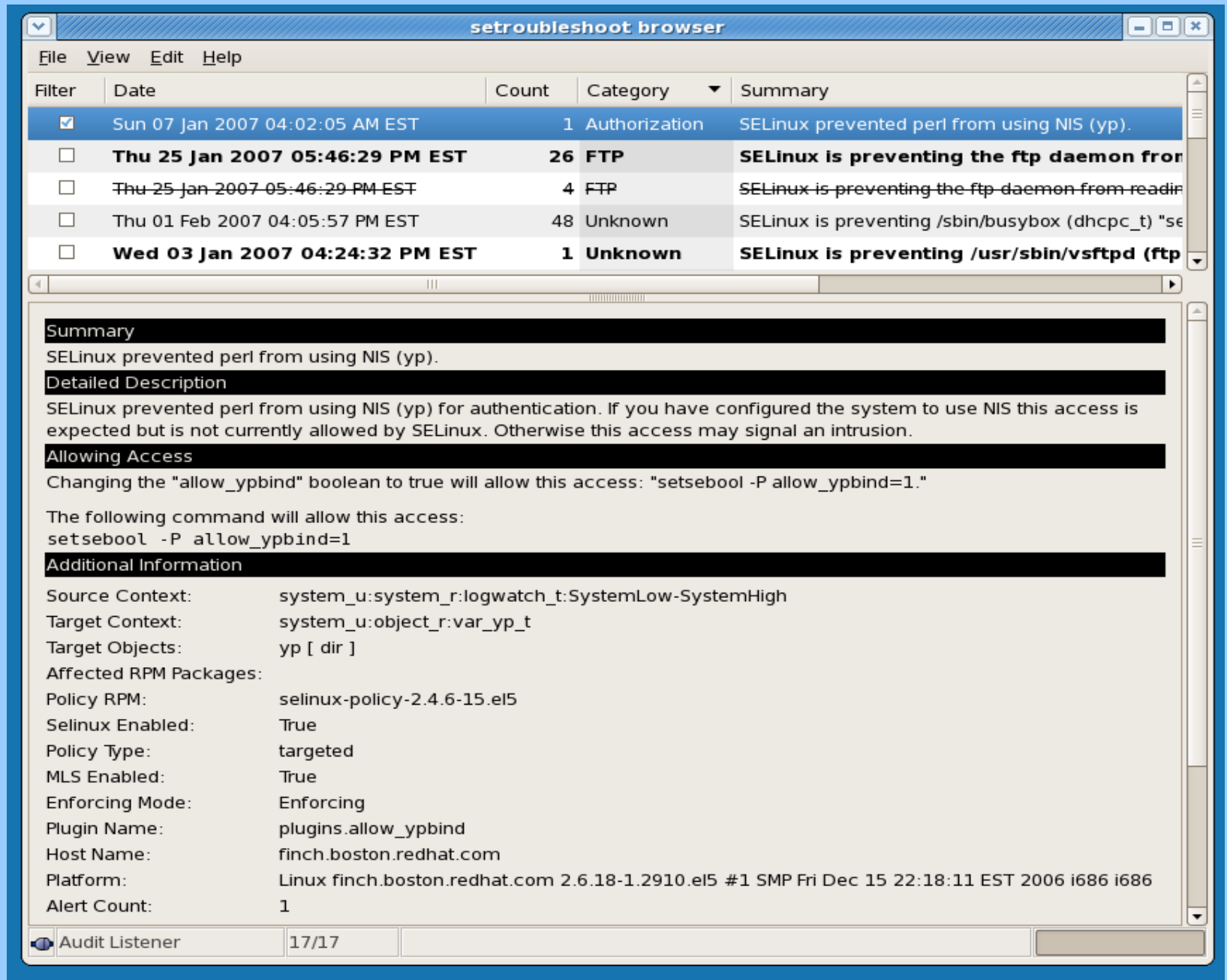
- Vital to make SELinux denials visible
  - Must be aware a denial occurred
  - users/developers are aided when denial is reported at the moment it occurs, helps correlate
- Denial must be comprehensible

# GUI Alert Notification

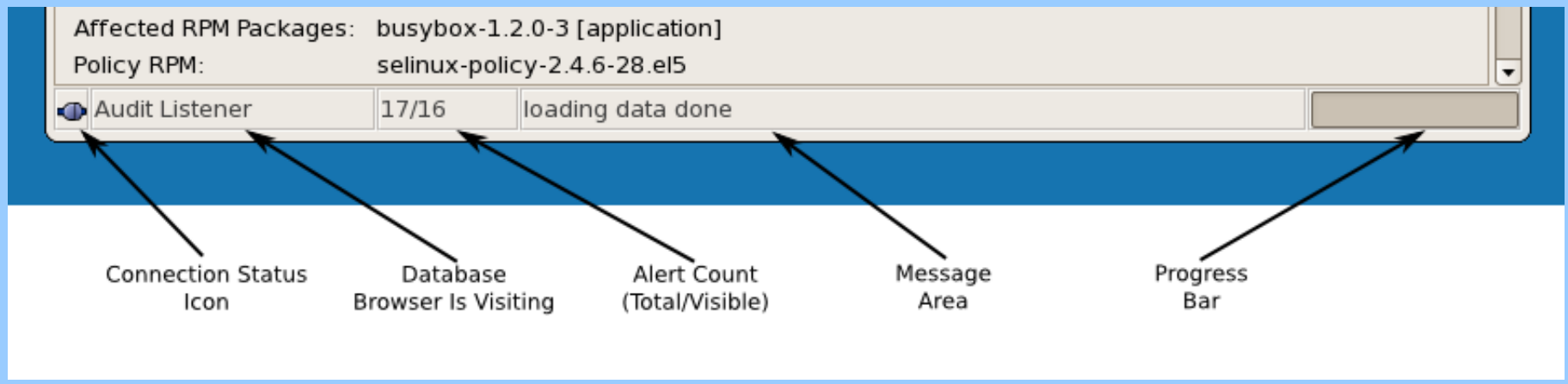
- Notification in real time
- Status icon appears (unviewed alert)
- Temporary balloon notification
- Click on icon to open alert browser



# Alert Browser



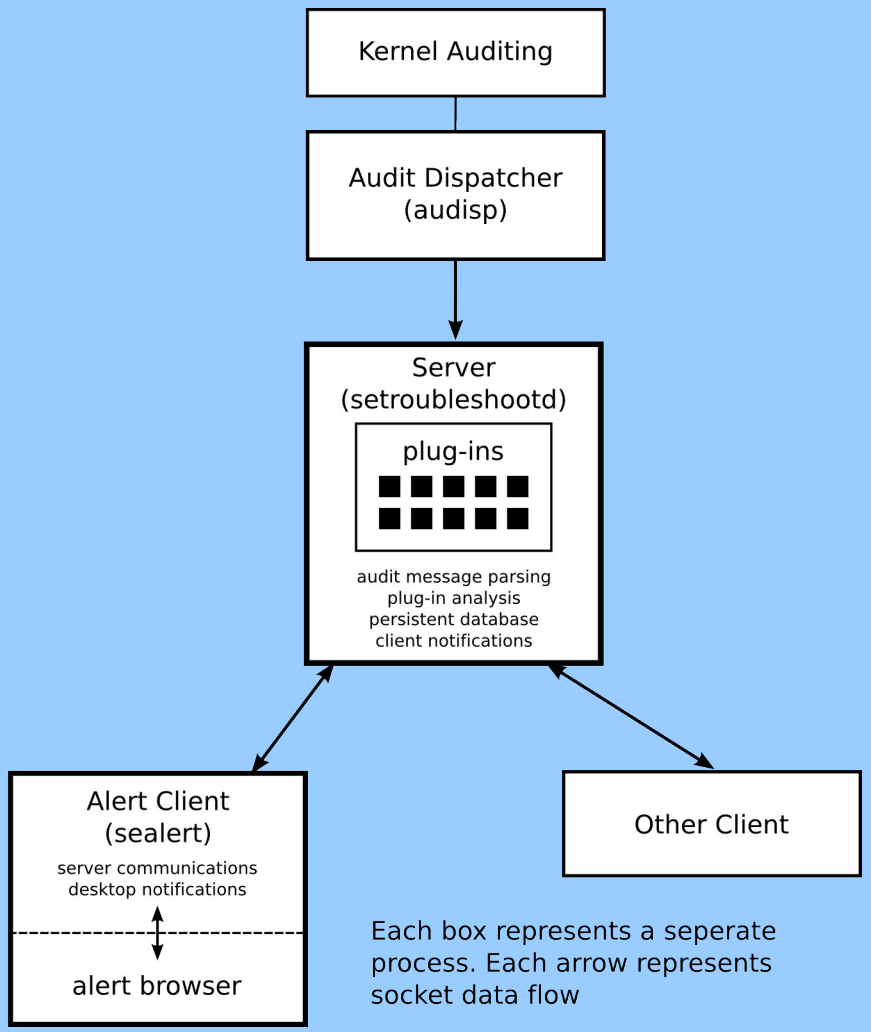
# Status Bar


Affected RPM Packages: busybox-1.2.0-3 [application]
Policy RPM: selinux-policy-2.4.6-28.el5
Audit Listener
17/16
loading data done
Connection Status Icon
Database Browser Is Visiting
Alert Count (Total/Visible)
Message Area
Progress Bar

# Two Major Components

- setroubleshootd (a.k.a server)
  - Real time audit connection
  - Persistent alert database for node
  - Permits connections for alert notification, queries
- sealert  (a.k.a. client)
  - GUI alert notification, browsing
  - Log file scanning (analysis)

# Setroubleshoot Architecture



# Audit Message Processing

- Receive audit message
- Synthesize AVC event from audit messages
- Place AVC event on analysis queue
- Dequeue AVC event, iterate over plugins
- Insert plugin report into target database
- Database change notification emitted

# Analysis

- Loadable plugins perform analysis
- Plugins have ordering precedence
- Plugin is provided a processed AVC object
- Plugin upon AVC match provides report
  - Summary, Description, Fix
- System environment query optional

# Alert Databases

- Alert databases store plugin reports (i.e. Alerts)
- Alert databases permit AVC events to be merged into existing alerts by signature
- Alert databases permit alerts to be grouped by node, log file, etc.
- An alert database is an XML document wrapped as an object with access and notification methods

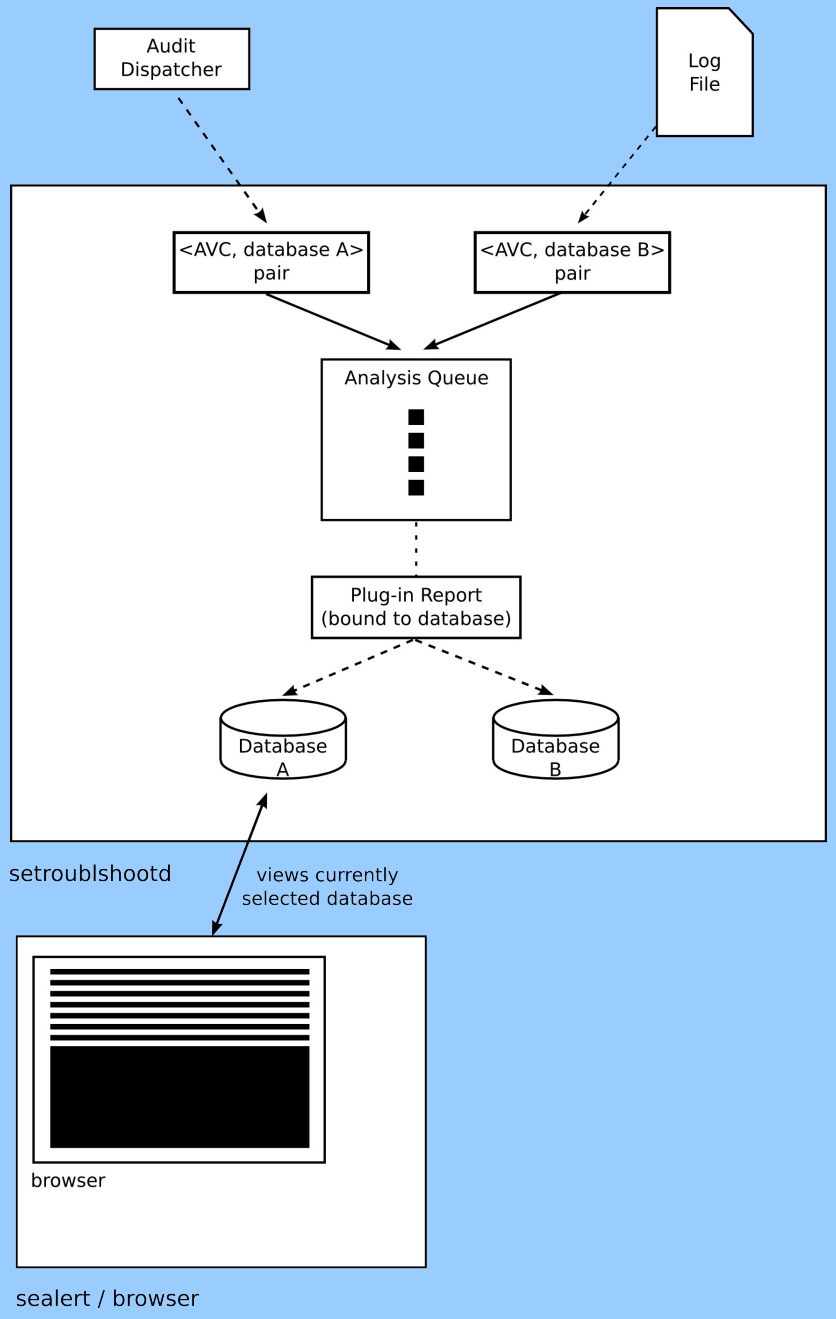
# Alert Signatures

- Alerts are keyed by signature
- Signatures are portable
- Signatures allow aggregation
- Plugin defines the signature
- A signature is the minimal AVC and environment properties needed to uniquely identify
- Signatures are small XML documents

# Log File Scanning

- Log files may be scanned & analyzed
  - In GUI browser
  - From command line (in text or HTML format)
- Final analysis produces a set of alerts
  - Each alert has unique signature
  - Alert occurrence count
  - Line number correlation
  - System environment info will be absent

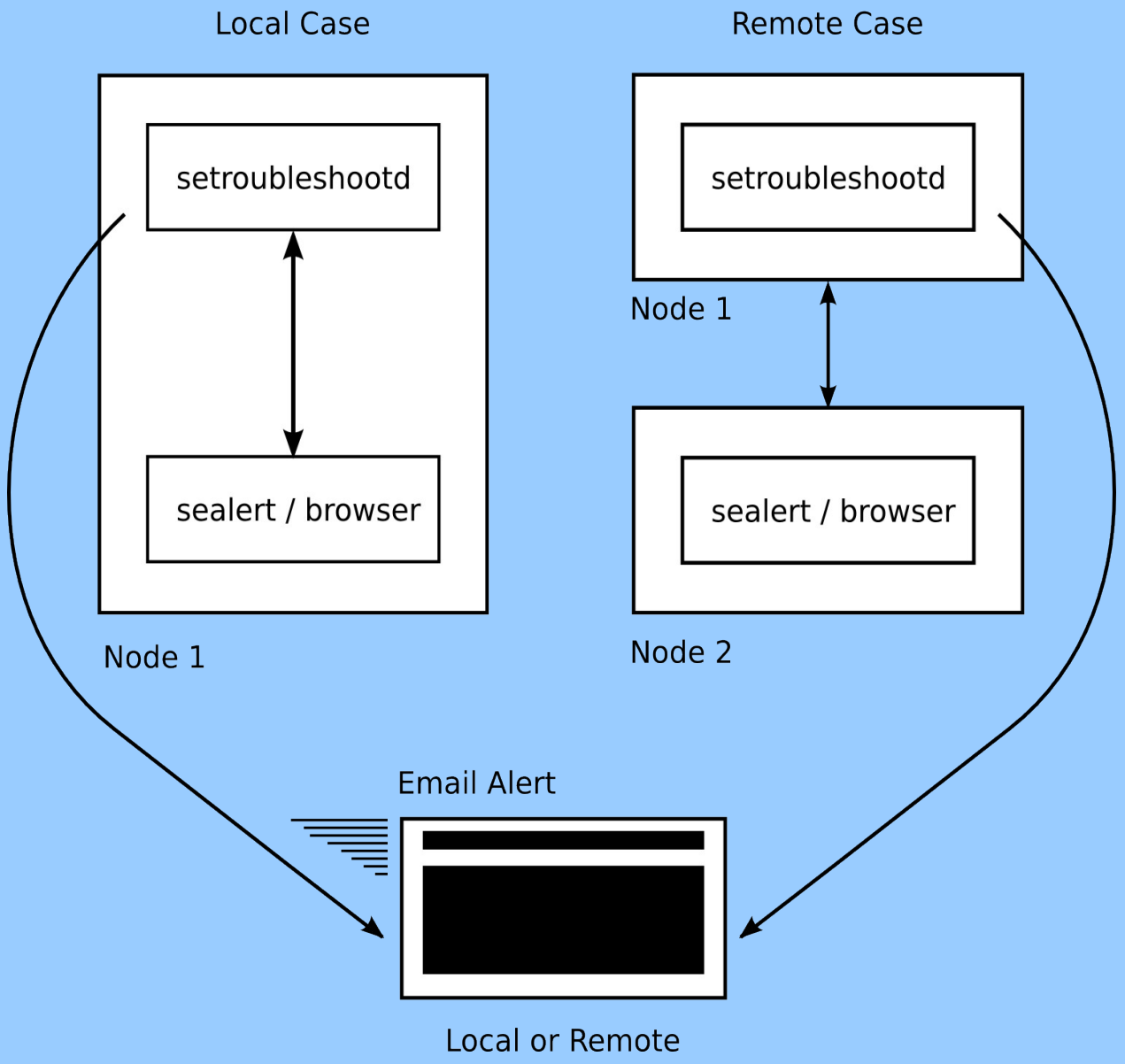
# Disposition of Alert Report



# Email Alerts

- Setroubleshootd can send email alerts
- List of email recipients and SMTP configurable
- Email alert can be suppressed via filtering
- Email is both plain text and HTML

# Operational Modes



# Conclusion

- An extensible tool has been built which aids
  - Developers
  - System Administrators
  - Users
- In recognizing AVC denials
  - Real time
  - From log files
  - Locally and Remotely
- Comprehending the denial & suggesting a solution

# Future Work

- Extending plugins
- Better integration with bug reporting
- Integrate with log aggregation
- Environment triggers (e.g. a new package is available which fixes an alert)