Monitoring and Alerting

All the things I've tried that didn't work, plus a few others.

By Aaron S. Joyner
Senior System Administrator
Google, Inc.

Blackbox vs Whitebox

Blackbox:

- Requires no participation of the monitored system
- Observes external functionality, "what the user sees"
- End to end test

Whitebox:

- Collects data intentionally provided by the target system
- Has more granular information about the system
- Can provide warning of problems before they occur
- Help with capacity planning, understanding of why things went (or are about to go) really sideways

Blackbox tests measure...

- Can you ping the webserver?
- Can you fetch a webpage from the webserver?
- Does it have the correct contents?

Whitebox monitoring collects...

- Ethernet interface statistics (packets/bytes sent/received)
- System Load (cpu, memory, uptime, disk usage)
- Daemon statistics (qps, uptime, cpu usage, version)

Historical vs Realtime Data

Historical Data

- "How many QPS did we see yesterday, before the DoS?"
- Useful for tracking traffic growth, finding trends, forensics
- Critical for evaluating your efforts long term (SLA/SLO)

Realtime Data

- Useful for evaluating current health
- Primarily drives alerting and troubleshooting
- Can make for great consoles

Monitoring vs Alerting vs Notification

Monitoring

- Collects the underlying data
- Organizes and displays data

Alerting

- Define the thresholds for "questionable" and "bad" performance
- Express dependencies, suppress duplicate alerts

Notification

- Something's wrong... who gets the page?
- Escalation: What if they don't respond?

Case Studies

(with some caveats)

Host Monitoring

Blackbox

- Reachability
- Remote Login (ssh, rdp)

Whitebox

- Uptime
- Load
- CPU Usage
- Memory
- Disk IO (per disk)
- Disk Errors (SMART)
- Network Interface stats

CGI Webapp (apache+php+mysql)

Blackbox

- Apache deliver static page
- "self-contained" PHP page
- PHP page w/ "simple" DB fetch
- PHP page w/ complicated DB fetch
- Latency of each above

Whitebox

- Apache
  - scrape the logs
  - scrape mod_status
- PHP
  - version, modules
- MySQL
  - 'show status'
  - 'show table status'
  - replication status
  - replication delay

Database (MySQL / Postgres)

Blackbox

- Can you execute a noop query? (select 1;)
- Can you execute a complicated query?
- How long does each query take?

Whitebox

- 'show status'
- 'show table status'
- 'show master status'
- 'show slave status'
- replication status
- replication delay

Mail System

Blackbox

- Test message delivery
- Test message latency
- Spam score of test msg

Whitebox

- Messages per second
- Num messages per queue
- CPU usage of spam scoring

Distributed indexed key/value pair storage system

- Stores X billion key/value pairs
- One frontend "index" server
- 15 backend "value" servers
- Index server looks up value from value server, returns to user

- Now let's design the monitoring...

Distributed indexed key/value pair storage system

Blackbox

- What can you think of?

Whitebox

- What can you think of?

Distributed indexed key/value pair storage system

Blackbox

- Store a new key
- Check key existence
- Retrieve value by key
- Request latency for each

Whitebox

- Index server uptime
- Value server uptime
- Index server qps
- Value server qps / instance
- Index server ram usage
- Value server disk usage
- Number of hash collisions
- Rate of collision resolution
- Replication qps
- Values per replication level

Distributed indexed key/value pair storage system

Blackbox

- Store a new key
- Check key existence
- Retrieve value by key
- Request latency for each

^^^^^^^^^^^^^^^^^^^^^^^^

These are effectively the SLOs for your SLA

Whitebox

- Index server uptime
- Value server uptime
- Index server qps
- Value server qps / instance
- Index server ram usage
- Value server disk usage
- Number of hash collisions
- Rate of collision resolution
- Replication qps
- Values per replication level

Alert Handling

How to get some sleep but keep everything running.

The usual way

- Page when a service, host, or network is down
- Send an email alert for everything else
- Limited console, usually "alerts firing" or "problems"

Basic Suggestions

- **No alerts via email!** (It doesn't scale.)
- **No alerts if they are not human actionable.**

- Pages for things that will break the SLA

- Tickets/Bugs for things that need a human to look at them, but can wait until morning.

- Consoles for displaying the first two categories, and graphs or list displays of anything else that might be relevant

- Auto-assign the tickets/bugs to humans

Pages vs Tickets

Page:

- Webserver not responding
- Page latency is > SLA
- Disk errors on the DB
- Test email undelivered >30m

Alert:

- Disk is over 90% capacity
- Test mail delivery took 5m
- High load on a machine

The Playbook

So you got a page...
now what?

Recipe for a good Playbook

- Problem description(s)
  - If necessary: how to narrow down the problem
- Severity
- Suggested resolution

If you add an alert, you're responsible for adding an entry for it.

If you work an alert from the playbook that isn't right, fix it.

If you have playbook entries with more than one resolution, you probably need more focused monitoring and corresponding alerts.

Remember, playbook customers are sleepy.

Questions?