Your bug tracker and you

Ben Cotton
SCaLE 19x
Housekeeping
Housekeeping

- Pronouns: he/him
- The usual disclaimers
- License: CC BY-SA 4.0
- Kind things: @FunnelFiasco
- Unkind things: /dev/null
- Questions at the end
Shameless plug
Agenda

- Why have a bug tracker?
- What to track
- Picking a tracker
- Triaging bugs
- Closing bugs
- Analyzing bugs
Not-agenda

• Fixing bugs
• Specific bug trackers
Why have a bug tracker?
Why have a bug tracker?

To track bugs, silly!
Why have a bug tracker?

Bug trackers are the cornerstone of building an open source community.
Why have a bug tracker?

- Communicate with users
- Coordinate with contributors
- Maintain a historical record
- Build a source of information about the project
What to track
What to track

- Type (e.g. “bug” or “RFE”)
- Status/state
- Closure type
- Impact
- Priority
What to track

- Version
- Component (if applicable)
- Platform (OS and architecture)
What to track

- Time (-stamps)
- Reporter
- Assignee
- more?
What to track

- Track these as separate fields, not bug content
- If you’re lucky, the tracker provides this already
Picking a tracker
Picking a tracker

- Authentication & accounts
- API libraries
- FOSS status
- Hosting
Picking a tracker

- Comment/attachment hiding
- Moderate accounts
- Private bugs/comments
Picking a tracker

- Customizable fields
- Metadata ACLs
- Links to version control, other bugs
- Bulk updates
- Milestones/epics
- Reporting
Picking a tracker

- Default assignees
- Multiple assignees
- Mentioning
- Watching
- Subscribing
Picking a tracker

Whatever you pick, you’ll use it as more than a bug tracker.
Triaging bugs
Triage questions (first pass)

- Is this a bug?
- Is it a duplicate?
- Is it a security bug?
- Is it filed in the right place?
- Is it reproducible?
- Does it say what happened?
Triage questions (prioritizing)

- Can physical harm result?
- Does it cause data loss or corruption?
- Does it permit unauthorized access?
- Does it reduce accessibility?
- Is the default configuration affected?
- Is there a *reasonable* workaround?
- Does the bug affect all platforms?
- Does it embarrass you?
Closing bugs
Closing bugs

You totally should!
To close or not to close

- Sends a clear signal of intent
- Makes the tracker easier to scan
- Makes people angry
- “I was going to work on that someday”
- Represents a catalog of known* issues
When to close a bug

- When it’s fixed
- If it’s a duplicate, invalid, or spam
- If it’s missing information
- If you can’t reproduce it
- If the version reaches end of life
- If you know you can’t or won’t fix it
How to close a bug

- Set and communicate a policy *ahead of time*
- Explain why
- Give instructions to reopen
- Ignore the complaints
Bug closure process

- Varies by closure type
- Don’t be too aggressive with “missing info” and “can’t reproduce” closures
Learn about your user community

- Easier than a well-constructed survey
- Learn the approximate
  - Usage by platform
  - Use of components
  - Adoption rate of new versions
Learn project trends

- Bugs over time
- Time to resolution
- Bug severity
What you can’t learn!

- Time-to-response
- Best (or worst) bug fixers
- Affiliations
Challenge your own assumptions

Bug severity

- unspecified
- urgent
- low
- high
- medium
How to analyze

- Look for trends that will cause a change in behavior
- Ignore duplicates (usually) and spam (always)
- Percentages versus raw numbers
How to analyze

• Jupyter Notebook and pandas
• R
• A spreadsheet, I guess?
Fedora Linux bugs

• “Exploring our bugs” at Nest With Fedora 2021
  – https://youtu.be/k9NWVYTXj_I
Let’s talk about it