

# Bug Triage for Ubuntu

By: Draycen DeCator (ddecator)

# What is this presentation for?

- This presentation is intended to introduce all of the information a person should need to triage bug reports.
- This presentation has a LOT of information. Please do not let this intimidate you. Much of this information is very basic and you may already know it. All of this information has been provided in order to give triagers a one-stop place to hopefully answer most of their questions.

# Why help triage bug reports?

- Developers are generally very busy fixing bugs, packaging their software, and writing code. As a result, they need help organizing their bug reports.
- It is a great way for Ubuntu users to get involved with the community, especially if they don't know much about programming.

# How does the process work?

- The Ubuntu BugSquad and BugControl teams follow specific guidelines in order to make working on general Ubuntu bugs more consistent.
- Members of the Ubuntu BugSquad team should become familiar with the guidelines so their work will match that of others.
- This presentation is meant to highlight the guidelines, but they can also be found here: <https://wiki.ubuntu.com/Bugs/HowToTriage>

# Where should you start?

- New BugSquad members should start wherever they are most comfortable working.
- Different types of work that can be done include: general triage of “New” reports, in-depth triage of reports for a package, and assigning packages.
- Each type will be covered by this presentation.

# General Triage of “New” Reports

- This section of triage involves working on bug reports that still have a “New” status.
- BugSquad members can learn a lot by working on these reports as it requires a familiarity with the triaging process.
- All “New” reports filed for Ubuntu can be found here: ["New" Reports](#)
- The “New” status is used for bugs that either have not been looked at before, or have new information that needs to be examined.

# General Triage

- When working on a new report, the first step should be to become familiar with the report by reading the description and the comments.
- After becoming familiar with the report, ask yourself if it is really talking about a bug or not. Some users file a bug report when they really mean to ask a question.
- Sometimes a bug report can really be a feature request. How to handle these bugs will be discussed later in the presentation.

# General Triage

- If the report is more of a question than a bug, you can convert the report into a question. When doing so, it is recommended you explain why, or use the [stock response](#).
- Launchpad provides a link on reports that allows for the easy conversion of a bug report into a question.





# General Triage

Firefox does not allow users to change the default search engine, unless the search bar is present 🐛

Ubuntu » "firefox" package » Bugs » **Bug #526453**


This bug affects you and 1 other person 🐛

Affects	Status	Importance	Assigned to	Milestone
▼  Mozilla Firefox 🐛	Confirmed 🐛	Unknown	 mozilla-bugs #331105	
▼  firefox (Ubuntu)	Triaged 🐛	Low 🐛	Unassigned 🐛	➕ Target to milestone

➕ Also affects project ➕ Also affects distribution 🔄 Nominate for release

Bug Description

This report is public 🐛

 Mark as duplicate  
➕ Convert to a question  
➕ Link a related branch  
➕ Link to CVE .....

🔴 Unsubscribe  
➕ Subscribe someone else


Subscribers

 Draycen DeCator  
 Robert Jordens

# General Triage

- If the report is indeed for a bug, then the next step is usually to check for duplicates.
- It is possible to search on Launchpad for duplicates, but the Launchpad search does not always show the reports you are looking for.
- Usually, a triager will get the best results from using a site search on a search engine.

# General Triage



site:bugs.launchpad.net <search terms>

# General Triage

- If you find that the report is a duplicate of another bug, then you should mark it as such.
- If there is not already a “Master” report, then the report with the most information should act as the main bug report.
- When there is a report that has already been triaged, then that report should be the “Master”

# General Triage

- When a bug has multiple reports, the “Master” report is the one that will be used for triaging. All of the duplicates will be marked as duplicates of the “Master” report.
- Sometimes reports will have “Master” in the name, usually when a bug is repeatedly reported. This is not always the case though.

# General Triage

- When determining if a bug report is a duplicate of another report, triagers must remember to consider the *cause* of the bug in the report.
- Two reports may describe similar behaviors, but have different causes. In this case, the two reports are for different bugs.
- Likewise, two reports may describe two different behaviors, but they have the same cause. In this case, they are reports of the same bug.

# General Triage

- When marking a bug as a duplicate, triagers should use the [stock response](#) and mark the bug as a duplicate.
- Remember that the stock response does not have the bug number in it, so that needs to be edited.

# General Triage

Firefox does not allow users to change the default search engine, unless the search bar is present

Ubuntu » "firefox" package » Bugs » **Bug #526453**

This bug affects you and 1 other person

Affects	Status	Importance	Assigned to	Milestone
Mozilla Firefox	Confirmed	Unknown	mozilla-bugs #331105	
firefox (Ubuntu)	Triaged	Low	Unassigned	Target to milestone

Also affects project Also affects distribution Nominate for release

Bug Description

This report is public

Mark as duplicate  
 Convert to a question  
 Link a related branch  
 Link to CVE

Unsubscribe  
 Subscribe someone else

## Subscribers

Draycen DeCator  
 Robert Jordens



# General Triage

- If a bug report is not a duplicate, then it needs to be examined for whether or not it has enough information for developers.
- It is extremely useful for bug reports to contain information added by Apport.
- Apport is a program that automatically collects information from a user's system depending on which package the bug is filed against.

# General Triage

- Most bugs are filed with Apport, so the information is already there.
- You can tell if Apport was used because a bug will contain:
  - The apport-bug tag
  - Information about the user's system in the description

# General Triage

- If a report has no Apport information, then triagers should usually ask the user to run apport-collect.
- A [stock response](#) is available for this. Remember, the bug number needs to be edited before you leave the comment.

# General Triage

- If a report has Apport information, then triagers should look to see if there is enough information for developers to work on the bug.
- How much information is needed depends on the software and the bug.
- If you are unsure whether a bug has enough information, you can always ask in #ubuntu-bugs

# General Triage

- If a bug report does not have enough information, there are two ways you can get more:
  - Reproduce the bug on your own system and add your findings to the report in a comment.
  - Ask the reporter to add more information.
    - [Stock responses](#) can be extremely helpful with this

# General Triage

- At this point, the triager can set the status of a report based on the following criteria:
  - Confirmed: You can reproduce the bug on your own system.
  - Incomplete: You have requested more information from the reporter.
  - Invalid: The problem is due to user error, or the user **stops responding** to information requests.

# General Triage

- Next, triagers should look to see if the bug has been filed upstream.
  - “Upstream” is used to describe packages that are developed by third party developers. They generally have their own bug tracking system and will not look at Launchpad.
  - Bug reports only need to be filed upstream if the bug is for a problem in the software itself. If the problem is downstream (or with Ubuntu), then it does not need to be filed upstream.

# General Triage

- If a bug needs to be filed upstream, then triagers should find out what bug tracker the upstream developers use.
- Once found, triagers should search to see if the bug has been reported already. If it has, then the Ubuntu bug report can be linked with that one. If it has not, then either you can file a report, or you can ask that the reporter does.



# General Triage

- It can be difficult to find upstream reports, but it is important that we do our best to not report duplicates. Upstream developers would rather be working on bug fixes than marking reports as duplicates, so triagers should spend some time trying different searches to make sure there are not reports already filed.

# General Triage

- Once an upstream report has been found, or a new one made, the downstream report should be linked to it.
- This can be done by clicking the “Also affects project” link on a report. Then, put the URL into the text field for “I already have the upstream URL” then click Continue.
- The reports will be linked, and changes to Importance and Status upstream will be shown on the Ubuntu report.

# General Triage

- If the report has enough information for developers, and has been filed upstream if necessary, then it is ready to be triaged.
- BugSquad triagers should determine the importance of the report, then request a BugControl member sets that importance and status on #ubuntu-bugs
- It helps to “clean” the title and description of the report if necessary. This makes it easier for developers to realize what the report is for and makes it easier for other triagers to find the report and possibly mark others as duplicates.

# General Triage

- Importance is determined by the following generalized criteria ([full details](#)):
  - Low: The problem is rare, does not effect functionality, does not effect a core application, and/or has an easy workaround.
  - Medium: Affects functionality, affects a core application, and/or is experienced by a moderate amount of users.
  - High: Severely affects functionality, affects essential hardware.
  - Critical: Severe impact for a lot of users.

# General Triage

- Sometimes a bug report is really a feature request. There are two ways to handle these reports:
  - If it is a minor change to the software, and is well defined, then it should be marked “Wishlist” by a BugControl member.
  - If it is for a large change, then the user should be directed to the Brainstorm page for Ubuntu using a [stock response](#).

# In-depth Reports

- Many BugSquad members find that they end up triaging bugs for just a few packages.
- By concentrating on a few packages, triagers become more aware of what information the developers need in bug reports.
- If a triager is interested in “adopting” a package, they can do so here:  
<https://wiki.ubuntu.com/BugSquad/AdoptPackage>

# Assigning Packages

- Many reports are submitted to Launchpad without a package being specified, or the wrong package was given so it needs re-assignment.
- Developers are not likely to see reports that are not assigned to their packages, so it is important that reports be assigned.
- Bug reports [without a package](#) tend to add up and need a lot of attention.
- If you see a report on there and you know what package it should be assigned to, please assign it to that package.

# General Tips

- Since many actions involved in bug triage are redundant, some tools have been made to help make many steps of triaging easier.
- One of the best is the set of Greasemonkey scripts available for Firefox users.
  - These scripts can be installed as a package from the following PPA:  
<https://launchpad.net/~gm-dev-launchpad/+archive/ppa>



# General Tips

- Many actions can be performed on a bug at one time if the drop-down menu is used on the report by clicking the arrow next to the package name. This helps to cut back on bug mail spam (emails are sent to users subscribed to the report whenever any actions happens on/to the report), and it saves triagers time.
- If firefox-lp-improvements is installed (from the PPA), then stock responses will be available from this extended view (and personal ones can be added).

# General Tips

Affects	Status	Importance	Assigned to	Milestone
Mozilla Firefox	Confirmed	Unknown	mozilla-bugs #331105	
firefox (Ubuntu)	Triaged	Low	Unassigned	Target to milestone

## Package

firefox (Choose...)

## Status Importance Milestone

Triaged Low (no value)

## Assigned to

- ☒ Nobody  
☐ Me  
☐ (Choose...)

## Comment on this change (optional)

{Obsolete Package} {4 Weeks (No Response)} {Invalid (No Response)} {Needs Apport (Firefox)} {Old Bug (Firefox)} {Test in Safe Mode (Firefox)} {New profile (Firefox)} [Collect it] [Vague] [Recreate Steps] [Kernel] [Sound] [ACPI] [Removable Device] [Printing] [Hardware Detection] [USB Storage] [X Autoconfig] [Xorg general] [Live CD Install] [Dist Upgrade] [GNOME Power Manager] [Network Manager] [Samba Client] [Samba Server] [Package assignment] [Unknown Fix] [Release+1 Testing] [A Duplicate] [Valgrind] [Upstreamed] [!Ubuntu Package] [Retrace Failed] [CoC] [Invalid Papercut] [Submit Upstream] [Added Bug Watch] [Use Storage Symptom] {Enable Apport (Firefox)} {Apport Failed (Firefox)} [+edit+] [+reload+]

Save Changes

# General Tips

- It is wise for triagers to subscribe to all of the bugs they work on. This will make it easier to go through them later. You will also get an email whenever a person leaves a comment on the report.
- This can be done by clicking the Subscribe link on the right side of the page, or by checking the Subscribe box on the extended view of a report (only visible when you are not subscribed).

# General Tips

- Triagers should make a habit of going through the bugs they are subscribed to on a regular basis, such as a certain day each week.
- This will also help you remember to mark reports as Invalid if necessary, and to see if there has been a response that you forgot to address.

# Thank you for the help!

- We always welcome new triagers, so everyone from the BugSquad thanks you for being willing to help!
- If you have any questions, then please feel free to ask them on #ubuntu-bugs