

D1 - Manual Assessment - Mozilla

We invite you to participate in a anonymous and voluntary questionnaire aimed at gathering your assessment of alternative answers given to technical questions made in Matrix channels regarding the Firefox project.

For each technical question, you will be presented a set of possible answers to this question. Initially, you should evaluate the alternative answers based on three criteria:

Helpfulness: the quality of providing assistance to address the question.

Comprehensiveness: the quality of including everything that is needed to answer the question.

Conciseness: the quality of including everything that is necessary for answering the question without using unnecessary words.

Additionally, there will be an open-ended question where you can share further insights to help us better understand the effectiveness of the various answers.

At the end of the survey, we would like to learn more about your background to better understand your experience and perspective.

* Indicates required question

Question 1 from Firefox Matrix.

<suppressed_user> Hi team 🙌,

Hope everybody is doing well.

We have a couple of [crash reports] (https://bugzilla.mozilla.org/show_bug.cgi?id=1873619) on Android, coming from `libutils.so`, with the below stack.

Is somebody familiar with this?

...

Reason: SIGSEGV / SEGV_MAPERR

0 libutils.so libutils.so@0xed0e0

1 boot.oat boot.oat@0x3844a0

2 ? @0x00006e90148652e4

...

Thanks in advance

1. Answer 1:

Hi 🙋, the crash reports indicate a segmentation fault (SIGSEGV) in libutils.so, which is a common library in Android's native code. The specific address libutils.so@0xed0 suggests an issue within that library, but without more context, it's challenging to pinpoint the exact cause. This could be due to memory corruption, invalid memory access, or a bug in the library itself. It would be helpful to gather more information, such as the Android version, device model, and steps to reproduce the crash. Additionally, checking for updates or patches for libutils.so might provide a solution.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

2. Answer 2:

The stacks look rather bad. Maybe <suppressed_user> would have some ideas of what could be going wrong. The devices the crashes are on all have the same manufacturer, so maybe that means something.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

3. Answer 3:

Hi <suppressed_user>,

The crash reports you're seeing are likely due to a segmentation fault in `libutils.so`. This is a known issue and has been reported in Bugzilla under [Bug 1873619] (https://bugzilla.mozilla.org/show_bug.cgi?id=1873619). You may want to follow the updates on that bug report for further information and potential fixes.

Check all that apply.

Helpful Comprehensive Concise

This

answer

is:

4. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

5. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.
-
-
-
-
-
-

Question 2 from Firefox Matrix.

<suppressed_user> Is there any really really simple place to store an int that isn't prefs? I want to store a timestamp preferably with more than seconds precision (so I don't want to just /1000 it)

6. Answer 1:

You can use the `nsIXULRuntime.data` object to store an int with more than seconds precision. This object allows you to store arbitrary data types, including integers, and is accessible across different parts of the codebase. This method avoids using preferences and provides the precision you need for a timestamp.

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

7. Answer 2:

You can use the nsIXULStore interface to store simple key-value pairs, including integers, with more precision than seconds. This interface is designed for storing small amounts of data and is simpler than using preferences. You can access it via JavaScript in your Firefox extension or application. Here's a quick example:

```
let xulStore = Components.classes["@mozilla.org/xul/xulstore;1"]  
    .getService(Components.interfaces.nsIXULStore);  
xulStore.setValue("your-namespace", "your-key", "your-timestamp-value");  
let timestamp = xulStore.getValue("your-namespace", "your-key");  
This allows you to store and retrieve your timestamp with the desired precision  
without using preferences.
```

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

8. Answer 3:

What's wrong with prefs? if your data is associated with a site you could use the content pref service

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

9. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

10. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.
-
-
-
-
-
-

Question 3 from Firefox Matrix.

<suppressed_user> Is there a helper function somewhere in the frontend to open a new browser window (BROWSER_CHROME_URL), and return a promise which resolves when gBrowser is available in that new window?

11. Answer 1:

Yes, you can use the BrowserTestUtils.openNewBrowserWindow function, which opens a new browser window and returns a promise that resolves when the new window's gBrowser is available.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

12. Answer 2:

Yes, you can use the `OpenBrowserWindow` function from `browser.js` to open a new browser window with `BROWSER_CHROME_URL`. It returns a promise that resolves when `gBrowser` is available in the new window.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

13. Answer 3:

BrowserTestUtils has some such thing but don't know the name off hand (on the phone sorry)

Check all that apply.

	Helpful	Comprehensive	Concise
This answer is:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

14. Please compare the answers:

Mark only one oval per row.

	Answer 1	Answer 2	Answer 3
--	-------------	-------------	-------------

**Which answer
provides the
most helpful
information?**

<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------	-----------------------

**Which answer
provides the
most
comprehensive
information?**

<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------	-----------------------

**Which answer
is the most
concise
without being
incomplete?**

<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------	-----------------------

**Which answer
do you prefer
to see in
practice?**

<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------	-----------------------

15. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.
-
-
-
-

Question 4 from Firefox Matrix.

Does someone here know what one can do about this? After upgrading to Firefox 121, it appears macOS is unwilling to allow Firefox to be started from the command-line, thus making e.g. unit tests via Karma and such impossible.

<https://github.com/karma-runner/karma-firefox-launcher/issues/328#issuecomment-1885113046>

macOS is not allowing the program to run? Not even for `–version`, or with no arguments.

```
$ /Applications/Firefox.app/Contents/MacOS/firefox-bin  
Killed: 9
```

16. Answer 1:

Based on vaguely related reports at

<https://apple.stackexchange.com/questions/258623/how-to-fix-killed-9-error-in-mac-os> and https://bugzilla.mozilla.org/show_bug.cgi?id=1704126, I'm guessing this has something to do with the way Mozilla signed the Firefox binary, and/or the way Firefox replaced its own binary (e.g. mv instead of cp or something like that)

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

17. Answer 2:

To address the issue of Firefox 121 being unable to start from the command line on macOS, try running the following command to remove the quarantine attribute: `xattr -d com.apple.quarantine /Applications/Firefox.app`. If this does not resolve the issue, consider downgrading Firefox or checking for updates on the related GitHub issue for any new fixes or workarounds.

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

18. Answer 3:

The issue you're encountering is likely due to macOS's Notarization and Hardened Runtime requirements, which prevent Firefox from being started from the command line without proper entitlements. To work around this, you can either disable System Integrity Protection (SIP), which is not recommended due to security risks, or use a local build or a try build of Firefox that does not have these restrictions. For local builds, ensure you follow the instructions for creating a debuggable build and signing it appropriately. You can find more details on building and signing local macOS builds in the Firefox developer documentation.

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

19. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

20. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 5 from Firefox Matrix.

<suppressed_user> Hello everyone,
I'm looking to perform a full rebranding of the Firefox application and package it as my
custom browser.

I have already built the firefox on Linux and Windows, and was able to make name changes
by changing MOZ_APP_NAME in several locations, it did changed the name, but the process
running is still firefox-default on Linux and Nightly on Windows

Could someone point me to the relevant documentation or provide guidance on how to
accomplish this?

Thank you!

21. Answer 1:

To fully rebrand Firefox and ensure that the process names reflect your custom
browser, you need to modify the branding files and possibly some build
configuration files. Start by updating the branding directory under browser/branding
with your custom icons, logos, and names. Ensure you change MOZ_APP_NAME
and MOZ_APP_DISPLAYNAME in the moz.build and confvars.sh files. Additionally,
you may need to update the application.ini file to reflect your custom names. For the
process names, you might need to modify the firefox.exe and related binaries
directly, which can be done by editing the source code where these names are
defined, typically in nsAppRunner.cpp and other related files. For more detailed
guidance, refer to the Mozilla Build Documentation (https://firefox-source-docs.mozilla.org/contributing/directory_structure.html) and the branding
documentation (<https://searchfox.org/mozilla-central/source/browser/branding>).

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

22. Answer 2:

I suck at explaining stuff, but I have my own rebranded fork, and you should be able to figure out what I did by looking at commit history <https://github.com/Eclipse-Community/r3dfox/commits/release/> Or wait for me to get around to 128 and I'll combine everything new into a single commit to look at

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

23. Answer 3:

To fully rebrand Firefox and ensure the process names reflect your custom browser, you need to modify the `MOZ_APP_DISPLAYNAME` and `MOZ_APP_BASENAME` variables in the relevant `moz.build` files and branding directories. Additionally, check `package-name.mk` and `application.ini` for any remaining references to "firefox" or "Nightly". For detailed guidance, refer to the [Firefox Build Documentation](<https://firefox-source-docs.mozilla.org/>).

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

24. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

25. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 6 from Firefox Matrix.

<suppressed_user> Can I enable a logger I'd normally use via `MOZ_LOG` for a mochitest suite on try too? do I just pass in the env somehow?

26. Answer 1:

Yes, you can enable a logger for a mochitest suite on try by setting the MOZ_LOG environment variable. To do this, modify the buildBrowserEnv method in the Python file for the test suite you are interested in and add the necessary environment variable settings. For example, you can add browserEnv["MOZ_LOG"] = "example_logger:3" to set the logging level for example_logger. Additionally, you can set MOZ_LOG_FILE to specify the log file path. This will ensure that the logs are captured and available for download from TreeHerder after the try run.

Check all that apply.

Helpful Comprehensive Concise

This answer is:

27. Answer 2:

Yes, with `--env` in the `mach try` invocation, IIRC I'd hope we have docs for this

Check all that apply.

Helpful Comprehensive Concise

This answer is:

28. Answer 3:

Yes, you can enable a logger via `MOZ_LOG` for a mochitest suite on try by setting the environment variable in the task configuration. You can do this by adding `mozharness` options in your `try` syntax, such as `try: -b do -p all -u mochitest --env MOZ_LOG=<module>:<level>`.

Check all that apply.

Helpful Comprehensive Concise

This

answer
is:

29. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

30. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 7 from Firefox Matrix.

<suppressed_user> My local mozilla-central builds are failing today with an `error[E0635]: unknown feature stdsimd`, on two separate machines, but I don't know what's changed recently to cause this anyone able to point me in the right direction?

31. Answer 1:

You're using nightly rust, is why

Check all that apply.

	Helpful	Comprehensive	Concise
This answer is:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

32. Answer 2:

The error error[E0635]: unknown feature stdsimd typically indicates that the Rust compiler version you're using does not support the stdsimd feature. To resolve this, ensure that your Rust toolchain is up-to-date by running `rustup update`. Additionally, verify that your `rust-toolchain` file in the mozilla-central directory specifies the correct Rust version. If the issue persists, you might want to check recent changes in the mozilla-central repository or consult the build documentation for any updates regarding Rust dependencies.

Check all that apply.

	Helpful	Comprehensive	Concise
This answer is:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

33. Answer 3:

The `error[E0635]: unknown feature stdsimd` indicates that the `stdsimd` feature is not recognized. This could be due to recent changes in the Rust compiler or the Mozilla codebase. To resolve this, ensure you are using the latest version of Rust by running `rustup update`, and also update your local mozilla-central repository by running `./mach bootstrap` followed by `./mach build`. If the issue persists, check the Mozilla build documentation or the recent commit history for any related changes.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

34. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

35. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 8 from Firefox Matrix.

<suppressed_user> Hello guys,
I have send a patch (my first here) for review more than two weeks ago but still didn't get any reviews. Is there anything that I did wrong? Should I add more reviewers? Ping people?

36. Answer 1:

Hi <suppressed_user>, it's common for reviews to take some time. You can gently ping the reviewers on the Bugzilla bug page or add one or two more relevant reviewers to get attention. Be sure to provide context and express your willingness to make any needed changes.

Check all that apply.

Helpful Comprehensive Concise

This answer is:

37. Answer 2:

It looks like you did the right thing on the patch. You could try pinging the reviewer here or needinfo them in bugzilla. We do also have a gfx-reviewer reviewer group but that's probably not quite the right thing for imagelib things.

Check all that apply.

Helpful Comprehensive Concise

This answer is:

38. Answer 3:

If your patch has been waiting for a review for more than two weeks, you should follow the "Getting attention" section guidelines. Add a comment to the bug saying 'review ping?', check the "Need more information from" box, and add the reviewer's name. If there is still no response within a day or two, you can ask for help on Matrix in the #introduction:mozilla.org or #developers:mozilla.org channels. Additionally, you can tag more reviewers or use review groups like #geckoview-reviewers in Phabricator to notify more people.

Check all that apply.

Helpful Comprehensive Concise

**This
answer
is:**

39. Please compare the answers:

Mark only one oval per row.

Answer	Answer	Answer
1	2	3

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

40. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 9 from Firefox Matrix.

<suppressed_user> I have a problem that's hard to debug -- `./mach test toolkit/components/extensions/test/xpcshell/test_ext_webRequest_responseBody.js` --- I get a timeout on the initial test in parallel. On the retry ('Retrying tests that failed when run in parallel') to run the xpcshell tests sequentially, I get an assertion. What I really want it to get an rr trace of the assertion, but if I use --debugger=rr, it never times out the initial test, so we never get to the one that fails.

Any suggestions?

```

```
12345670:01.02 SUITE_START: xpcshell - running 2 tests
0:01.02 TEST_START: xpcshell-
e10s.toml:toolkit/components/extensions/test/xpcshell/test_ext_webRequest_responseBod
y.js
0:01.02 TEST_START: xpcshell-
remote.toml:toolkit/components/extensions/test/xpcshell/test_ext_webRequest_responseB
ody.js
5:01.02 TEST_END: TIMEOUT
5:01.02 INFO xpcshell return code: None
5:01.02 TEST_END: TIMEOUT
5:01.02 INFO xpcshell return code: None
````
```

(and then on the retry after a timeout I get the IOThread assertion)

41. Answer 1:

You can try running the test with `--log-tbpl -` to get more detailed logs, which might help identify the root cause of the timeout and the assertion failure. Additionally, consider using `rr record` to capture the trace of the initial run, even if it doesn't fail, and then use `rr replay` to debug the subsequent assertion. This way, you might capture more context leading up to the failure.

Check all that apply.

Helpful Comprehensive Concise

This

answer

is:

42. Answer 2:

To debug the issue, try running the test with the `--sequential` flag to avoid parallel execution, which might help isolate the problem. Additionally, you can use the `MOZ_DEBUG_CHILD_PROCESS=1` environment variable to pause the child process and attach a debugger to it. If the test still times out, consider adding more logging to identify where it hangs. If you need an rr trace, ensure the test runs long enough to capture the failure by adjusting timeouts or adding delays. Finally, verify if adding `firefox-appdir = browser` to the `xpcshell.toml` manifest resolves any resource access issues.

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

43. Answer 3:

Maybe rr chaos mode helps? Something like `--debugger-args='record --chaos'`

Check all that apply.

Helpful Comprehensive Concise

This
answer
is:

44. Please compare the answers:

Mark only one oval per row.

| Answer | Answer | Answer |
|--------|--------|--------|
| 1 | 2 | 3 |

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

45. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Question 10 from Firefox Matrix.

<suppressed_user> Hi, I am trying to debug an issue with fenix on my Pixel 6a where it crashes. I wanted to run a debug build to get a stack trace. I set up the build and it completed without errors, but when I start the intent it immediately fails with no output (the log in android studio is just connected/disconnected from target VM). Any advice on how to debug this? Is my phone haunted?

46. Answer 1:

To debug the issue with Fenix crashing on your Pixel 6a, first ensure that USB debugging is enabled on your device and that you have the necessary permissions. Use adb logcat to capture detailed logs from your device, which can provide more insight into the crash. You can also try running the app directly from the command line using ./gradlew :app:installFenixDebug and then start it with adb shell am start -n org.mozilla.fenix.debug/[org.mozilla.fenix.debug.App](#). If the issue persists, check for any missing dependencies or misconfigurations in your build setup. Additionally, ensure that your Android SDK and Java paths are correctly set. If none of these steps resolve the issue, consider reaching out to the Fenix development community for further assistance.

Check all that apply.

Helpful Comprehensive Concise

This answer is:

47. Answer 2:

Maybe check on GeckoView. There's nothing in the logcat?

Check all that apply.

Helpful Comprehensive Concise

This

answer
is:

48. Answer 3:

To debug the issue with fenix crashing on your Pixel 6a, try enabling verbose logging in Android Studio by going to "Run" > "Edit Configurations" and setting the log level to "Verbose". Additionally, make sure your phone's developer options have "USB Debugging" enabled, and try running `adb logcat` from the terminal to capture detailed logs. If the app crashes before any log output, consider adding breakpoints in the initialization code to narrow down the crash point.

Check all that apply.

Helpful Comprehensive Concise

This

answer
is:

49. Please compare the answers:

Mark only one oval per row.

| Answer | Answer | Answer |
|--------|--------|--------|
| 1 | 2 | 3 |

**Which answer
provides the
most helpful
information?**

**Which answer
provides the
most
comprehensive
information?**

**Which answer
is the most
concise
without being
incomplete?**

**Which answer
do you prefer
to see in
practice?**

50. Regarding the answer you prefer to see in practice, we encourage you to emphasize all particular aspects that are (ir)relevant in the answer(s) and aspects you think it missed.

Characterization

Thank you for your valuable collaboration! In this section, we would like to learn more about your background to better understand your experience and perspective.

51. How confident were you in assessing the answers?

Mark only one oval.

1 2 3 4 5

Not Very confident

52. How many years of professional experience do you have? *

Mark only one oval.

Less than 1 year

1-3 years

4-7 years

8-10 years

More than 10 years

53. How many years have you been contributing to Mozilla projects? *

Mark only one oval.

Less than 1 year

1-3 years

4-7 years

8-10 years

More than 10 years

54. What is your gender?

55. How familiar are you with answering technical questions in Matrix channels?

Mark only one oval.

1 2 3 4 5

Not Very familiar

56. How do you contribute to support newcomers in Mozilla projects?

This content is neither created nor endorsed by Google.

Google Forms

