




☆ Starred by 6 users

Owner:

 [ricea@chromium.org](mailto:ricea@chromium.org)  
OOO until January 5th

CC:

[adetaylor@chromium.org](mailto:adetaylor@chromium.org)  
[keta...@chromium.org](mailto:keta...@chromium.org)  
 [cros-its-team@google.com](mailto:cros-its-team@google.com)  
[c...@samy.pl](mailto:c...@samy.pl)  
[adetaylor@google.com](mailto:adetaylor@google.com)  
 [ketakid@google.com](mailto:ketakid@google.com)  
[vsavu@google.com](mailto:vsavu@google.com)  
[sporeba@google.com](mailto:sporeba@google.com)  
[stefanoduo@google.com](mailto:stefanoduo@google.com)

Status:

Verified (Closed)

Components:

[Internals>Network](#)

Modified:

26 days ago

Backlog-Rank:

----

Editors:

----

EstimatedDays:

----

NextAction:

2020-11-09

OS:

Linux, Android, Windows, Chrome, Mac, Fuchsia

Pri:

1

Type:

[Bug-Security](#)

Hotlist-Merge-Review

Security\_Impact-Stable

reward-NA

Security\_Severity-High

allpublic

CVE\_description-submitted

M-87

Target-86

Target-87

Merge-Rejected-86

merge-merged-4240

merge-merged-86

merge-merged-4280

merge-merged-87

LTC-Merged-86

LTS-Security-86

Release-0-M87

merge-merged-4240\_112

reward\_to-code\_at\_samy.pl

**Issue 1145680: Ports 5060 and 5061 should be blocked**

Reported by [ricea@chromium.org](mailto:ricea@chromium.org) on Wed, Nov 4, 2020, 2:47 PM EST Project Member

 Code

Some NAT devices scan packets on port 5060 and will dynamically create port forwards if they think they see a valid SIP request.

There is a proof-of-concept of how to exploit this. See <https://samy.pl/slipstream>.

The Fetch standard is also adding port 5061 to the bad ports list, so Chrome should follow suit: <https://github.com/whatwg/fetch/issues/1108>

Comment 1 by [ricea@chromium.org](mailto:ricea@chromium.org) on Wed, Nov 4, 2020, 2:51 PM EST Project Member

**Labels:** Security\_Impact-Stable Security\_Severity-Low OS-Android OS-Chrome OS-Fuchsia OS-Linux OS-Mac OS-Windows  
**Components:** Internals>Network

Comment 2 by [ricea@chromium.org](mailto:ricea@chromium.org) on Wed, Nov 4, 2020, 2:51 PM EST Project Member

How can I remove the view restriction? The issue is already public.

Comment 3 by [adetaylor@chromium.org](mailto:adetaylor@chromium.org) on Wed, Nov 4, 2020, 3:03 PM EST Project Member

**Cc:** [adetaylor@chromium.org](mailto:adetaylor@chromium.org)

Comment 4 by [adetaylor@chromium.org](mailto:adetaylor@chromium.org) on Wed, Nov 4, 2020, 3:04 PM EST Project Member

**Labels:** allpublic  
I can't remove it either. I am exploring what's going on.

Comment 5 by [adetaylor@chromium.org](mailto:adetaylor@chromium.org) on Wed, Nov 4, 2020, 3:04 PM EST Project Member

OK that worked.

Comment 6 by [bugdroid](#) on Wed, Nov 4, 2020, 9:10 PM EST Project Member

The following revision refers to this bug:  
<https://chromium.googlesource.com/chromium/src.git/+90d1302aec437166b383eabc08af741bf2477ea8>  
  
commit [90d1302aec437166b383eabc08af741bf2477ea8](#)  
Author: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Date: Thu Nov 05 02:09:29 2020  
  
Add ports 5060 and 5061 to the restricted list

Some NAT devices examine traffic on port 5060 to look for a valid SIP message. If they find one, they will forward a port back to the origin host. A carefully crafted HTTP request can trick these NAT devices into forwarding an arbitrary port. See <https://samy.pl/slipstream> for more details on the attack and sample code.

Block port 5060 for HTTP. Out of an abundance of caution, and to match the Fetch standard (<https://github.com/whatwg/fetch/pull/1109>), also block port 5061 (SIP over TLS).

Also reduce the whitespace before protocol description comments. This was insisted on by clang-format and is not worth fighting.

**BUG-4445689**

Change-Id: I3a556fbbb4dc6099caa4418addaf1e89bf254ae3  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2519174>  
Reviewed-by: Matt Menke <mmenke@chromium.org>  
Commit-Queue: Adam Rice <ricea@chromium.org>  
Cr-Commit-Position: refs/heads/master@{#824254}

[modify] [https://crrev.com/90d1302aec437166b383eabc08af741bf247ea8/net/base/port\\_util.cc](https://crrev.com/90d1302aec437166b383eabc08af741bf247ea8/net/base/port_util.cc)

**Comment 7** by [ricea@chromium.org](#) on Wed, Nov 4, 2020, 9:22 PM EST Project Member  
**Status:** Fixed (was: Assigned)

**Comment 8** by [adetaylor@google.com](#) on Thu, Nov 5, 2020, 11:18 AM EST Project Member  
**Labels:** -Security\_Severity-Low

In an e-mail ricea@ said he needed help assessing the severity here, so I'm clearing the severity field such that the sheriffs look at it.

**Comment 9** by [adetaylor@google.com](#) on Thu, Nov 5, 2020, 11:37 AM EST Project Member  
**Labels:** Security\_Severity-High reward\_to-code\_at\_samy.pl

I've asked the current sheriff (kenrb@) to have a look (it won't show up on the normal dashboard as this is marked Fixed).

Meanwhile here are my views on severity. Ken may disagree and override.

First of all it's a terrific write-up. Obviously I haven't attempted to reproduce the problem, but if it works as intended, I think it deserves High severity. The consequence is that a properly remote attacker running a malicious website can access ports on the local network which the user might reasonably believe is firewalled off from the world. That sounds High to me,

One could argue whether it's a bug in Chrome or in the NAT gateways or just emergent behavior of the complexity of the modern internet, but either way, I think we should merge the fix with urgency commensurate with High severity.

Meanwhile ricea@ - do you believe blocking port 5060/5061 is sufficient here? The article hints that there may be other affected services. Do you believe that a blocklist of ports is sufficient overall for this class of attacks or do we need to involve the WebRTC team in some other solution?

As this was publicly disclosed, I don't believe this is subject to VRP reward, but I'd still like to credit the reporter in the Chrome release notes, so adding appropriate label.

**Comment 10** by [sheriffbot](#) on Thu, Nov 5, 2020, 11:39 AM EST Project Member  
**Labels:** reward-potential

**Comment 11** by [adetaylor@google.com](#) on Thu, Nov 5, 2020, 11:41 AM EST Project Member  
(As this is now marked High severity and fixed, I anticipate Sheriffbot will add Merge-Request-87 during its next daily run, and I think that's good.)

**Comment 12** by [adetaylor@google.com](#) on Thu, Nov 5, 2020, 11:42 AM EST Project Member  
**Labels:** reward-na

**Comment 13** by [sheriffbot](#) on Thu, Nov 5, 2020, 12:48 PM EST Project Member  
**Labels:** M-86 Target-86

Setting milestone and target because of Security\_Impact=Stable and high severity.

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

**Comment 14** by [kenrb@chromium.org](#) on Thu, Nov 5, 2020, 12:50 PM EST Project Member  
I can see an argument that it should only be Sev-Medium based on the set of preconditions for this to work, but it also seems right to merge it back since it can be quite bad in some specific situations and Samy's post has gotten some public attention and it will likely be in attacker toolkits very soon.

Based on that I'd say Sev-High is appropriate.

> "The article hints that there may be other affected services."

Possibly someone should look into ALG and see what other ports are relevant. Some are ports that are already restricted, but I see that RTSP is mentioned (554) which is not.

**Comment 15** by [sheriffbot](#) on Thu, Nov 5, 2020, 2:19 PM EST Project Member  
**Labels:** Merge-Request-87 Merge-Request-86

Requesting merge to stable M86 because latest trunk commit (824254) appears to be after stable branch point (800218).

Requesting merge to beta M87 because latest trunk commit (824254) appears to be after beta branch point (812852).

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

**Comment 16** by [sheriffbot](#) on Thu, Nov 5, 2020, 2:22 PM EST Project Member  
**Labels:** -Merge-Request-87 Merge-Review-87 Hotlist-Merge-Review

This bug requires manual review: We are only 11 days from stable.  
Before a merge request will be considered, the following information is required to be added to this bug:

1. Does your merge fit within the Merge Decision Guidelines?  
- Chrome: [https://chromium.googlesource.com/chromium/src.git/+master/docs/process/merge\\_request.md#when-to-request-a-merge](https://chromium.googlesource.com/chromium/src.git/+master/docs/process/merge_request.md#when-to-request-a-merge)  
- Chrome OS: <https://goto.google.com/cros-release-branch-merge-guidelines>
2. Links to the CLs you are requesting to merge.
3. Has the change landed and been verified on ToT?
4. Does this change need to be merged into other active release branches (M-1, M+1)?
5. Why are these changes required in this milestone after branch?
6. Is this a new feature?
7. If it is a new feature, is it behind a flag using finch?

Chrome OS Only:  
8. Was the change reviewed and approved by the Eng Prod Representative? See Eng Prod ownership by component: <http://go/cros-engprodcomponents>

Please contact the milestone owner if you have questions.  
Owners: benmason@ (Android), bindusuvama @ (iOS), cindyb@ (ChromeOS), lakpamarthy@ (Desktop)

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

**Comment 17** by [ricea@chromium.org](#) on Thu, Nov 5, 2020, 3:18 PM EST Project Member

1. Mostly.  
Full automated unit test coverage: The code in question is well tested. The specific ports added here are not tested, but we wouldn't gain much meaningful coverage by doing so. I can write a unit test anyway if it helps.  
Deployed in Canary for at least 24 hours: Not yet. Given the urgency I am requesting a merge before it is baked, and I will merge once it is.  
Safe merge: It's complicated. The risk to Chrome stability is essentially zero. However, by design this change blocks access to server ports that were previously allowed. Servers on those ports will no longer be accessible. This is mitigated by the fact that running HTTP(S) servers on those ports is not common practice.
2. <https://chromium-review.googlesource.com/c/chromium/src/+2519174>
3. Yes.
4. As a high severity security issue, it should be merged to stable.
5. Publicly disclosed high-severity security issue.
6. No

**Comment 18** by [adetaylor@chromium.org](#) on Thu, Nov 5, 2020, 4:20 PM EST Project Member

**Labels:** -Merge-Request-86 Merge-Rejected-86  
**NextAction:** 2020-11-09

Thanks for the very helpful description of the pros and cons of merging. We should either merge this into M87 now for the first M87 release (in 11 days), or wait two weeks then merge into the first M87 security refresh. Given that this is public it does seem plausible that people will try to exploit this, so I'm inclined to merge now. But I'm going to think again on Monday then approve or reject the merge.

We should not merge this into M86. It's unlikely that we'll do another M86 refresh before the first M87 release, and if we do, we wouldn't want to rush this out to that extent anyway.

**Comment 19** by [ricea@chromium.org](#) on Thu, Nov 5, 2020, 4:44 PM EST Project Member

#9 I have been in contact with the WebRTC folk and they are working on mitigations from several directions.

**Comment 20** by [ricea@chromium.org](#) on Sun, Nov 8, 2020, 11:49 PM EST Project Member

The change has now baked in canary over the weekend.

**Comment 21** by [adetaylor@chromium.org](#) on Mon, Nov 9, 2020, 2:00 AM EST Project Member

**Labels:** -Merge-Review-87 Merge-Approved-87

Approving merge to M87, branch 4280, assuming there's no sign of trouble from Canary.

**Comment 22** by [bugdroid](#) on Mon, Nov 9, 2020, 4:18 AM EST Project Member

**Labels:** -merge-approved-87 merge-merged-87 merge-merged-4280

The following revision refers to this bug:  
<https://chromium.googlesource.com/chromium/src.git/+dbb0452e69a49e803e0e4cbb6921d5ccad338716>

commit [dbb0452e69a49e803e0e4cbb6921d5ccad338716](#)  
Author: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Date: Mon Nov 09 09:16:54 2020

Add ports 5060 and 5061 to the restricted list

Some NAT devices examine traffic on port 5060 to look for a valid SIP message. If they find one, they will forward a port back to the origin host. A carefully crafted HTTP request can trick these NAT devices into forwarding an arbitrary port. See <https://samy.pl/slipstream> for more details on the attack and sample code.

Block port 5060 for HTTP. Out of an abundance of caution, and to match the Fetch standard (<https://github.com/whatwg/fetch/pull/1109>), also block port 5061 (SIP over TLS).

Also reduce the whitespace before protocol description comments. This was insisted on by clang-format and is not worth fighting.

~~BUG-4446600~~

(cherry picked from commit [90d1302aec437166b383eabc08af741bf2477ea8](#))

Change-Id: I3a556fbbb4dc6099caa4418addaf1e89bf254ae3  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2519174>  
Reviewed-by: Matt Menke <[mmenke@chromium.org](mailto:mmenke@chromium.org)>  
Commit-Queue: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Cr-Original-Commit-Position: refs/heads/master@{#824254}  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2525474>  
Reviewed-by: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Cr-Commit-Position: refs/branch-heads/4280@{#1247}  
Cr-Branched-From: [ea420fb963f9658c9969b6513c56b8f47efa1a2a](#)-refs/heads/master@{#812852}

[modify] [https://crrev.com/dbb0452e69a49e803e0e4cbb6921d5ccad338716/net/base/port\\_util.cc](https://crrev.com/dbb0452e69a49e803e0e4cbb6921d5ccad338716/net/base/port_util.cc)

**Comment 23** by [adetaylor@google.com](#) on Thu, Nov 12, 2020, 3:37 PM EST Project Member

**Labels:** Merge-Request-86

It looks like M87 may be significantly delayed, so adding a merge request for 86 for consideration after all.

**Comment 24** by [adetaylor@google.com](#) on Thu, Nov 12, 2020, 8:08 PM EST Project Member

**Labels:** -Merge-Request-86

Let's not take this into M86 even if M87 will be a bit later. There is (low) risk of actual breakage here and I don't think there's quite enough time to have found out.

**Comment 25** by [adetaylor@google.com](#) on Mon, Nov 16, 2020, 10:40 AM EST Project Member

**Labels:** Release-0-M87

**Comment 26** by [adetaylor@google.com](#) on Mon, Nov 16, 2020, 11:07 AM EST Project Member

**Cc:** c...@samy.pl

[code@samy.pl](mailto:code@samy.pl) - hi - I'm planning to credit this in the Chrome release notes to @SamyKamkar - I hope that's OK with you.

**Comment 27** by [samy@samy.pl](#) on Mon, Nov 16, 2020, 12:37 PM EST

Thanks!

Comment 28 by adetaylor@google.com on Mon, Nov 16, 2020, 12:46 PM EST Project Member

**Labels:** CVE-2020-16022 CVE\_description-missing

Comment 29 by jayds...@gmail.com on Thu, Nov 19, 2020, 12:32 PM EST

Hi, are there any work-arounds to unblock 5060/5061?

Comment 30 by adetaylor@chromium.org on Thu, Nov 19, 2020, 12:54 PM EST Project Member

jaydstein@ I don't know the answer to your question, but I think we'd be really pleased to hear more about your use-case and what's been broken by this change, if you don't mind giving a bit more detail?

Comment 31 by ricea@chromium.org on Thu, Nov 19, 2020, 2:54 PM EST Project Member

#29 You can use the --explicitly-allowed-ports command-line argument. For example --explicitly-allowed-ports=5060,5061

Comment 32 by bugdroid on Tue, Nov 24, 2020, 6:21 AM EST Project Member

**Labels:** merge-merged-4240\_112

The following revision refers to this bug:

<https://chromium.googlesource.com/chromium/src.git/+67416a5ca83334a4765f871d0b064581b7c982e4>

commit 67416a5ca83334a4765f871d0b064581b7c982e4

Author: Adam Rice <ricea@chromium.org>

Date: Tue Nov 24 11:19:52 2020

Add ports 5060 and 5061 to the restricted list

Some NAT devices examine traffic on port 5060 to look for a valid SIP message. If they find one, they will forward a port back to the origin host. A carefully crafted HTTP request can trick these NAT devices into forwarding an arbitrary port. See <https://samy.pl/slipstream> for more details on the attack and sample code.

Block port 5060 for HTTP. Out of an abundance of caution, and to match the Fetch standard (<https://github.com/whatwg/fetch/pull/1109>), also block port 5061 (SIP over TLS).

Also reduce the whitespace before protocol description comments. This was insisted on by clang-format and is not worth fighting.

[BUG=11445690](#)

(cherry picked from commit 90d1302aec437166b383eabc08af741bf24f7ea8)

(cherry picked from commit dbb0452e69a49e803e0e4cbb6921d5ccad338716)

Change-Id: I3a556fbbb4dc6099caa4418addaf1e89bf254ae3

Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2519174>

Reviewed-by: Matt Menke <mmenke@chromium.org>

Commit-Queue: Adam Rice <ricea@chromium.org>

Cr-Original-Original-Commit-Position: refs/heads/master@{#824254}

Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2525474>

Reviewed-by: Adam Rice <ricea@chromium.org>

Cr-Original-Commit-Position: refs/branch-heads/4280@{#1247}

Cr-Original-Branched-From: ea420fb963f9658c9969b6513c56b8f47efa1a2a-refs/heads/master@{#812852}

Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2555119>

Reviewed-by: Achuth Bhandarkar <achuth@chromium.org>

Reviewed-by: Victor-Gabriel Savu <vsavu@google.com>

Commit-Queue: Artem Sumaneev <asumaneev@google.com>

Cr-Commit-Position: refs/branch-heads/4240\_112@{#46}

Cr-Branched-From: 427c00d3874b6abcf4c4c2719768835fc3ef26d6-refs/branch-heads/4240@{#1291}

Cr-Branched-From: f297677702651916bbf65e59c0d4bbd4ce57d1ee-refs/heads/master@{#800218}

[modify] [https://crrev.com/67416a5ca83334a4765f871d0b064581b7c982e4/net/base/port\\_util.cc](https://crrev.com/67416a5ca83334a4765f871d0b064581b7c982e4/net/base/port_util.cc)

Comment 33 by vsavu@google.com on Thu, Nov 26, 2020, 4:35 AM EST Project Member

**Labels:** LTS-Security-86 Merge-Request-86-LTS

Comment 34 by asumaneev@google.com on Thu, Nov 26, 2020, 10:55 AM EST Project Member

Cc: keta...@chromium.org cros-lts-team@google.com

ketakid@: could you take a look at merge request for LTS?

Comment 35 by sheriffbot on Thu, Nov 26, 2020, 12:21 PM EST Project Member

**Labels:** -M-86 M-87 Target-87

Comment 36 by mgar...@gmail.com on Wed, Dec 2, 2020, 10:26 AM EST

#30 - since you requested information about a use case (which I think will be resolved for my team by #31 or by reverse proxying, but just as an FYI) - the deaf/hard of hearing community has a lot of legacy infrastructure that uses valid SIP messages on these ports to coordinate softphone calls between users in that community, and occasionally between members of that community, sign interpreters, and hearing users. Much of this is done with thick clients today, but this impacts an effort I'm working on to create a more modern, more platform agnostic client based on chromium for this user group.

I'm happy to chat about this use case more if you're interested, just let me know.

Comment 37 by samy@samy.pl on Wed, Dec 2, 2020, 11:52 AM EST

#36 that sounds really interesting, is there somewhere I can learn more about this project? Is it communicating using HTTP/HTTPS/WebSockets over those ports?

Comment 38 by ketakid@google.com on Wed, Dec 2, 2020, 12:50 PM EST Project Member

**Labels:** Merge-Approved-86-LTS

Comment 39 by asumaneev@google.com on Wed, Dec 2, 2020, 3:29 PM EST Project Member

**Labels:** -Merge-Request-86-LTS LTC-Merged-86

Comment 40 by bugdroid on Wed, Dec 2, 2020, 3:29 PM EST Project Member

**Labels:** merge-merged-4240 merge-merged-86

The following revision refers to this bug:

<https://chromium.googlesource.com/chromium/src.git/+bbc6ab5bb49c54c58591d052e5f6e383363bb143>

commit bbc6ab5bb49c54c58591d052e5f6e383363bb143

Author: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Date: Wed Dec 02 20:26:52 2020

Add ports 5060 and 5061 to the restricted list

Some NAT devices examine traffic on port 5060 to look for a valid SIP message. If they find one, they will forward a port back to the origin host. A carefully crafted HTTP request can trick these NAT devices into forwarding an arbitrary port. See <https://samy.pl/slipstream> for more details on the attack and sample code.

Block port 5060 for HTTP. Out of an abundance of caution, and to match the Fetch standard (<https://github.com/whatwg/fetch/pull/1109>), also block port 5061 (SIP over TLS).

Also reduce the whitespace before protocol description comments. This was insisted on by clang-format and is not worth fighting.

**BUG-4445699**

(cherry picked from commit [90d1302aec437166b383eabc08af741bf24f7ea8](#))

(cherry picked from commit [dbb0452e69a49e803e0e4cbb6921d5ccad338716](#))

Change-Id: I3a556fbbb4dc6099caa4418addaf1e89bf254ae3  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2519174>  
Reviewed-by: Matt Menke <[mmenke@chromium.org](mailto:mmenke@chromium.org)>  
Commit-Queue: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Cr-Original-Original-Commit-Position: refs/heads/master@{#824254}  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2525474>  
Reviewed-by: Adam Rice <[ricea@chromium.org](mailto:ricea@chromium.org)>  
Cr-Original-Commit-Position: refs/branch-heads/4280@{#1247}  
Cr-Original-Branched-From: [ea420fb963f9658c9969b6513c56b8f47efa1a2a](#)-refs/heads/master@{#812852}  
Reviewed-on: <https://chromium-review.googlesource.com/c/chromium/src/+2560585>  
Reviewed-by: Achuth Bhandarkar <[achuth@chromium.org](mailto:achuth@chromium.org)>  
Reviewed-by: Victor-Gabriel Savu <[vsavu@google.com](mailto:vsavu@google.com)>  
Commit-Queue: Artem Sumaneev <[asumaneev@google.com](mailto:asumaneev@google.com)>  
Cr-Commit-Position: refs/branch-heads/4240@{#1474}  
Cr-Branched-From: [f297677702651916bbf65e59c0d4bbd4ce57d1ee](#)-refs/heads/master@{#800218}

[modify] [https://crrev.com/bbc6ab5bb49c54c58591d052e5f6e383363bb143/net/base/port\\_util.cc](https://crrev.com/bbc6ab5bb49c54c58591d052e5f6e383363bb143/net/base/port_util.cc)

**Comment 41** by [sheriffbot](#) on Mon, Dec 7, 2020, 12:13 PM EST Project Member

Cc: [adetaylor@google.com](mailto:adetaylor@google.com) [ketakid@google.com](mailto:ketakid@google.com) [vsavu@google.com](mailto:vsavu@google.com)

This issue has been approved for a merge. Please merge the fix to any appropriate branches as soon as possible!

If all merges have been completed, please remove any remaining Merge-Approved labels from this issue.

Thanks for your time! To disable nags, add the Disable-Nags label.

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

**Comment 42** by [sheriffbot](#) on Thu, Dec 10, 2020, 12:13 PM EST Project Member

This issue has been approved for a merge. Please merge the fix to any appropriate branches as soon as possible!

If all merges have been completed, please remove any remaining Merge-Approved labels from this issue.

Thanks for your time! To disable nags, add the Disable-Nags label.

For more details visit <https://www.chromium.org/issue-tracking/autotriage> - Your friendly Sheriffbot

**Comment 43** by [aashu...@chromium.org](#) on Tue, Dec 15, 2020, 1:48 PM EST Project Member

**Status:** Verified (was: Fixed)

Verified on Sarien M88-13597.15.0

Steps:

1. run `python3 -m http.server 5060` & `python3 -m http.server 5061` to serve local files via browser.
2. check `http://0.0.0.0:5060` throws ERR\_UNSAFE\_PORT error. (Same of 5061 port)
3. Other unblocked ports should be able to serve local files.

@ricea Thanks for directions.

**Comment 44** by [adetaylor@google.com](mailto:adetaylor@google.com) on Thu, Jan 7, 2021, 1:52 PM EST Project Member

**Labels:** -CVE\_description-missing CVE\_description-submitted

**Comment 45** by [janag...@google.com](mailto:janag...@google.com) on Tue, Jan 19, 2021, 1:44 PM EST Project Member

**Labels:** -Merge-Approved-86-LTS

**Comment 46** by [adetaylor@google.com](mailto:adetaylor@google.com) on Wed, Jan 20, 2021, 6:57 PM EST Project Member

**Labels:** -reward-potential external\_security\_report

**Comment 47** by [stefanoduo@google.com](mailto:stefanoduo@google.com) on Tue, Nov 22, 2022, 9:45 AM EST (26 days ago) Project Member

**Labels:** Cronet