

[New issue](#)[Jump to bottom](#)

[bug]: Fail to chain sync on testnet3 & mainnet errors relating to: script witness item is larger than the max allowed size #7002

✓ Closed rsafier opened this issue on Oct 9 · 24 comments · Fixed by [#7004](#)

Labels bug P0

rsafier commented on Oct 9 • edited ▼

Background

Fail to chain sync on testnet3 .

The node does show it is up to the tip of the chain , but remains unsynced due to errors

```
2022-10-09 18:42:39.886 [ERR] LNWL: Unable to process chain reorg: unable to get block
0000000000000032fcf519b61aad2b966348e3f2d27687b26277933cc9881965: readScript: script witness item
is larger than the max allowed size [count 396669, max 11000]
```

....

```
2022-10-09 18:46:18.715 [ERR] LNWL: Unable to complete chain rescan: readScript: script witness
item is larger than the max allowed size [count 396669, max 11000]
```

...

```
2022-10-09 21:36:23.272 [ERR] LNWL: Unable to process chain reorg: unable to get block
0000000000000014820a254fbcddcce582df2194014c1e3b3f5ecd9259fce663: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
```

Your environment

- 0.15.0-beta commit=v0.15.0-beta
- Linux ip-10-21-2-70 5.13.0-1031-aws #35~20.04.1-Ubuntu SMP Mon Jun 13 22:30:30 UTC 2022 x86_64 x86_64 GNU/Linux
- backend bitcoind v23.0.0

Steps to reproduce

Unknown

Expected behaviour

"synced_to_chain": true

Actual behaviour

"synced_to_chain": false with following errors in logs:

```
2022-10-09 18:42:39.886 [ERR] LNWL: Unable to process chain reorg: unable to get block
000000000000032fcf519b61aad2b966348e3f2d27687b26277933cc9881965: readScript: script witness item
is larger than the max allowed size [count 396669, max 11000]
```

....

```
2022-10-09 18:46:18.715 [ERR] LNWL: Unable to complete chain rescan: readScript: script witness
item is larger than the max allowed size [count 396669, max 11000]
```

...

```
2022-10-09 21:36:23.272 [ERR] LNWL: Unable to process chain reorg: unable to get block
000000000000014820a254fbcddcce582df2194014c1e3b3f5ecd9259fce663: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
```



  rsafier added **bug** **needs triage** labels on Oct 9

RandyMcMillan commented on Oct 9

Contributor



benthecarman commented on Oct 9

Contributor

Looks like Ind/btcd has a bug in their taproot implementation. In BIP342:

Script size limit The maximum script size of 10000 bytes does not apply. Their size is only implicitly bounded by the block weight limit.[\[9\]](#)

I am surprised this wasn't caught by tests. Are they not using the [static test vectors](#)?



HamishMacEwan commented on Oct 9

Having this with LND Version v0.15.1 on mainnet:

```
2022-10-10 10:27:48.835 [INF] LTND: Waiting for chain backend to finish sync, start_height=757924
2022-10-10 10:27:49.535 [ERR] LNWL: Unable to complete chain rescan: readScript: script witness
item is larger than the max allowed size [count 33970, max 11000]
```



benthecarman commented on Oct 9

Contributor

Got the same error on mainnet:

```
2022-10-09 16:56:50.707 [ERR] LNWL: Unable to process chain reorg: unable to get block
000000000000000000000000400a35a007e223a7fb8a622dc7b5aa5eaace6824291fb: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
```

rsafier commented on Oct 9

Author

I also also seeing on mainnet v0.15.0:

```
2022-10-09 21:56:48.549 [ERR] LNWL: Unable to process chain reorg: unable to get block
000000000000000000000000400a35a007e223a7fb8a622dc7b5aa5eaace6824291fb: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
```

ExImpius commented on Oct 9

Same

fotongit commented on Oct 9 • edited ▼

Answer: a transaction came in the Blockchain that LND was not able to manage, thus all nodes went down at the same block

Contributor

```
2022-10-09 21:56:55.488 [ERR] LNWL: Unable to process chain reorg: unable to get block
000000000000000000000000400a35a007e223a7fb8a622dc7b5aa5eaace6824291fb: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
```



Contributor

<https://mempool.space/tx/7393096d97bfee8660f4100ffd61874d62f9a65de9fb6acf740c4c386990ef73>

Contributor

<https://mempool.space/tx/7393096d97bfee8660f4100ffd61874d62f9a65de9fb6acf740c4c386990ef73>

 3

```
2022-10-09 22:20:06.866 [INF] LTND: Waiting for chain backend to finish sync, start_height=2350195
2022-10-09 22:20:07.828 [INF] LNWL: Started rescan from block
00000000000000027e44f015d141df1aef3794ac4b0b7303ca1039c6baa35a4c (height 2350075) for 0 addresses
2022-10-09 22:20:08.093 [ERR] LNWL: Unable to complete chain rescan: readScript: script witness
item is larger than the max allowed size [count 396669, max 11000]
```

seeing this printed over and over in bitcoind debug.log

```
2022-10-09T22:30:27Z ThreadRPCServer method=getblockheader user=bitcoinrpc
2022-10-09T22:30:28Z ThreadRPCServer method=getblockchaininfo user=bitcoinrpc
```

bnonni commented on Oct 9

This seems to be the tx that can't be parsed

<https://mempool.space/tx/7393096d97bfee8660f4100ffd61874d62f9a65de9fb6acf740c4c386990ef73>

Posted about here: <https://twitter.com/brqgoo/status/1579216353780957185>

if this is a mainnet tx, why is it impacting nodes running on testnet?

  **bentthecarman** mentioned this issue on Oct 9

October topics [austinbitdevs/austinbitdevs.com#11](https://austinbitdevs.com/#11)

🔒 Closed

bentthecarman commented on Oct 9

Contributor



This seems to be the tx that can't be parsed

mempool.space/tx/7393096d97bfee8660f4100ffd61874d62f9a65de9fb6acf740c4c386990ef73

Posted about here: twitter.com/brqgoo/status/1579216353780957185

if this is a mainnet tx, why is it impacting nodes running on testnet?

A similar transaction happened on testnet

  **rsafier** changed the title ~~[bug]: Fail to chain sync on testnet3 errors relating to: script witness item is larger than the max allowed size~~ [bug]: Fail to chain sync on testnet3 & mainnet errors relating to: script witness item is larger than the max allowed size on Oct 9

bnonni commented on Oct 9

This seems to be the tx that can't be parsed

mempool.space/tx/7393096d97bfee8660f4100ffd61874d62f9a65de9fb6acf740c4c386990ef73

Posted about here: twitter.com/brqgoo/status/1579216353780957185

if this is a mainnet tx, why is it impacting nodes running on testnet?

A similar transaction happened on testnet

makes sense. figured he must have tried it on testnet first. but the testnet block referenced doesn't appear to have a tx at the same scale as the one on mainnet

<https://mempool.space/testnet/block/0000000000000027e44f015d141df1aeff3794ac4b0b7303ca1039c6baa35a4c>

spyhuntergenral commented on Oct 9

ahh, few, thought i broke my node for a second .

benthecarman commented on Oct 9

Contributor

The bad tx on testnet is

<https://mempool.space/testnet/tx/44692bc2da73192cd0b89bc7a43c0ce43578f6b3567bc945e46e6952e8ec5ca5>

bnonni commented on Oct 9

Ahh, thanks. I see, its rescanning starting at block height 2350075 meaning the tx is in the next block. Got it. Well damn, I finally got my testnet node up-and-running and was about to make some taro assets and now this 😞

jblachly commented on Oct 9

Contributor

Bug is actually in btcd code:

<https://github.com/btcsuite/btcd/blob/fc36cb25a4bdc6d989f9161552e7b0fe08b02939/wire/msgtx.go#L105-L109>





saubyk added the `p0` label on Oct 9

brqgoo commented on Oct 9 • edited ▾

With BIP-342, the maximum script size limit of 10000 bytes no longer applies. The witness script size is only implicitly bounded by the block weight limit.

https://github.com/bitcoin/bips/blob/master/bip-0342.mediawiki#cite_ref-9-0

Roasbeef commented on Oct 9

Member

Hey y'all, thanks for bringing this to our attention. I've identified the issue in the `btcd` wire parsing library, which led to this incident.

AFAICT, the consensus code wasn't the issue here, it was instead that the wire parsing library was erroneously still enforcing a prior check to limit witness sizing left over from segwit v0.

I am surprised this wasn't caught by tests. Are they not using the [static test vectors](#)?

`btcd` is/was using this test vectors. The issue here is that the code that parsed the witnesses for these test vectors isn't the same code that's used to read blocks off the *wire*. When a new block comes in, we fetch the raw block then attempt to decode it, which triggered this issue.



29

indomitorum commented on Oct 9 • edited ▾

Same problem. Have stopped the node now until further notice. 0.15.1

Logs when starting the node

Oct 10 00:34:36 indomitusbtc Ind[1922]: 2022-10-10 00:34:36.014 [INF] LTND: Waiting for chain backend to finish sync, start_height=757941

Oct 10 00:34:36 indomitusbtc Ind[1922]: 2022-10-10 00:34:36.399 [INF] LNWL: Started rescan from block 0000000000000000000000007477b90ff0ea53cdd1db88e03799af18ff58df0cebaa7 (height 757921) for 2056 addresses

Oct 10 00:34:36 indomitusbtc Ind[1922]: 2022-10-10 00:34:36.685 [ERR] LNWL: Unable to complete chain rescan: readScript: script witness item is larger than the max allowed size [count 33970, max 11000]



saubyk removed the `needs triage` label on Oct 9

Roasbeef commented on Oct 9 • edited ▾

Member

The fix to the btcd wire parsing logic can be found here: [btcsuite/btcd#1896](https://github.com/btcsuite/btcd/pull/1896)

This should be safe to apply to those running *btcd* nodes, which'll allow them to resume validating the main chain (the block was accepted as this wasn't a *consensus* issue persay).

Once this passes CI and a few more sniff checks, we'll issue a hotfix release for Ind: 0.15.2. This release will *only* contain the dependency update to the wire parsing library.



🔗 Roasbeef added a commit to Roasbeef/Ind that referenced this issue on Oct 9

 build: update to version of btcd with wire parsing bug fix ...

536c94c

🔗  Roasbeef mentioned this issue on Oct 9

build: update to version of btcd with wire parsing bug fix #7004

 Merged

🔗  LightningLuke mentioned this issue on Oct 9

REF: Ind major bug reported #7002 (affecting all nodes) - Unable to deserialize block: readScript: script witness item is larger than the max allowed size [count 33970, max 11000]
mynodebtc/mynode#702

 Closed

🔗 Roasbeef added a commit to Roasbeef/Ind that referenced this issue on Oct 9

 build: update to version of btcd with wire parsing bug fix ...

b3dc56f

🔗 Roasbeef added a commit to Roasbeef/Ind that referenced this issue on Oct 9

 build: update to version of btcd with wire parsing bug fix ...

a502c38

chappjc commented on Oct 9 • edited ▾

Is a bump for neutrino needed too? I think a consumer can hoist the required version themselves, but they might have to require btc directly to do so.



LND suddenly inactive rootzoll/raspibltz#3350

✓ Closed

Update LND to v0.15.2 (URGENT - hot fix) raspibolt/raspibolt#1083

Merged

4 tasks



LND not syncing due to a bug (urgent hotfix) [mynodebtc/mynode#704](#)

 Closed

okeygo commented on Oct 10 • edited ▼

I confirm that my LND nodes are stuck at block 757921 despite I can send and receive Stas normally through established channels, I am proceeding to implement fix.

```
2022-10-10 08:19:21.786 [ERR] LNWL: Unable to process chain reorg: unable to get block
00000000000000000000400a35a007e223a7fb8a622dc7b5aa5eaace6824291fb: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
2022-10-10 08:19:22.258 [ERR] LNWL: Unable to process chain reorg: unable to get block
00000000000000000000400a35a007e223a7fb8a622dc7b5aa5eaace6824291fb: readScript: script witness item
is larger than the max allowed size [count 33970, max 11000]
2022-10-10 08:20:02.063 [INF] CRTR: Processed channels=0 updates=198 nodes=11 in last
1m0.000466268s
2022-10-10 08:20:22.643 [INF] DISC: Broadcasting 321 new announcements in 18 sub batches
2022-10-10 08:21:02.063 [INF] CRTR: Processed channels=0 updates=234 nodes=8 in last 59.999372778s
```



[bug]: Unable to complete chain rescan: readScript: script witness item is larger than the max allowed #7009

✓ Closed



seberm mentioned this issue on Oct 10

Ind: 0.15.1-beta -> 0.15.2-beta NixOS/nixpkgs#195337

Merged



vertiond mentioned this issue on Oct 10

Update LND to v0.15.2-beta to fix stopped mainnet nodes getumbrel/umbrel#1519

Closed



a5an0 mentioned this issue on Oct 10

new fun transaction janoside/btc-rpc-explorer#477

Closed



brh28 mentioned this issue on Oct 10

Socratic Seminar #0 Topics NashBitDevs/nashbitdevs.org#1

Closed



nepet added a commit to nepet/peerswap that referenced this issue on Oct 10



gomod: Update to lnd v0.15.2-beta ...

d88023c



nepet mentioned this issue on Oct 10

gomod: Update to lnd v0.15.2-beta ElementsProject/peerswap#142

Merged



VajraOfIndra mentioned this issue on Oct 11

[FEATURE REQUEST] Add a LND chain/graph sync status raspibolt/raspibolt-pulse#6

Closed



sputn1ck mentioned this issue on Oct 11

not enough witness outputs to create funding transaction, need xxx BTC only have 0 BTC available #7023

Closed

  fiatjaf mentioned this issue on Oct 12

plugin-trustedcoin getblock error nbd-wtf/trustedcoin#14

 Closed

 This was referenced on Oct 13

[feature]: Remove ErrSynchronizingChain from returned error whitelist #7034

 Closed

Remove ErrSynchronizingChain #7039

 Merged

  reez mentioned this issue on Oct 15

Socratic Seminar #1 Topics NashBitDevs/nashbitdevs.org#2

 Closed

  otech47 mentioned this issue 23 days ago

Explore possible refactors or additional test cases in btcd/wire/msgtx.go btcsuite/btcd#1915

 Open

 SachinMeier pushed a commit to SachinMeier/lnd that referenced this issue 22 days ago



build: update to version of btcd with wire parsing bug fix ...

f43db41

  GoVulnBot mentioned this issue 19 days ago

x/vulndb: potential Go vuln in github.com/lightningnetwork/lnd: CVE-2022-44797
golang/vulndb#1098

 Closed

Assignees

No one assigned

Labels

bug P0

Projects


None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

 **build: update to version of btcd with wire parsing bug fix**
Roasbeef/lnd

16 participants

