QuorumDMS / **ftp-srv** Public

Code   **Issues** 21   Pull requests 12   Discussions   Actions   Projects   ...

New issue

Jump to bottom

Windows: User can escape from root directory #167

Open   **yetanothernickname** opened this issue on Jul 17, 2019 · 20 comments · Fixed by #224

Labels   bug   **security**   **windows**

---

yetanothernickna... commented on Jul 17, 2019 • edited

Windows, default File System, root directory set in login event callback. User can browse parent directory using /../../ in URL.
Example:
`root: 'X:\\Project\\Storage\\User'`

URL `ftp://127.0.0.1/../../` becomes a command `CWD \/../../`

At line 30 in fs.js we have `nodePath.resolve('X:\Project\Storage\User', '.\\\..\..\')`

So _resolvePath() returns

```
{
  clientPath: '\\..\..\',
  fsPath: 'X:\Project'
}
```

---

trs commented on Jul 19, 2019   Contributor

This must be a bug on windows. I'll have to investigate.

There is a specific test to ensure users cannot escape the root directory: https://github.com/trs/ftp-srv/blob/master/test/fs.spec.js#L65

The `clientPath` is passed through `normalize` which would resolve any `.` and `..`.
For example, in the test case, attempting to change directory to `../../../../../../../../../../..` would result in a `clientPath` (the path the client is actually shown) as `/`. Whereas the `fsPath` (the actual path on the server) would resolve to the root.

---

trs added bug **help wanted** **input needed** labels on Jul 19, 2019

trs mentioned this issue on Jul 19, 2019

Various Issue Fixes #168
Merged

---

yetanothernickna... commented on Jul 19, 2019   Author

Notice the backslash in the CWD command. I can't explain it, but..

```
const path = require('path');
console.log(path.normalize('\\/../../../../'));
```

Linux:
`../../../`

Windows:
`\\..\..\`

---

trs commented on Jul 20, 2019 • edited   Contributor

Trying this on windows I can't recreate it using Filezilla, Firefox, or Edge. What client are you using?

But in the next PR I have removed the added separator when joining paths, this *should* solve your issue.

---

yetanothernickna... commented on Jul 20, 2019   Author

Firefox. Demo
I have a suspicion that this module is not designed to work with URLs...

---

trs commented on Aug 8, 2019   Contributor

Thanks for the demo video. The attached PR (#168) should address this issue. I'll work on getting it released soon.

---

yetanothernickna... commented on Aug 11, 2019   Author

You can see how #168 works in this video :)

trs commented on Aug 12, 2019 Contributor

@yetanothernickname You used #168 for that video?

yetanothernickna... commented on Aug 12, 2019 Author

@trs Yes, with new fsPath()

matt-forster commented on Dec 8, 2020 Contributor

I'd like to see if we can confirm this is still the case with the changes we made because it shouldn't be.

n-timofeev commented on Dec 9, 2020

@forstermatth I was able to reproduce it on `4cd88b1`

👍 1

matt-forster commented on Dec 9, 2020 Contributor

@n-timofeev Thanks, I will take a look this week.

matt-forster pinned this issue on Dec 9, 2020

matt-forster added a commit that referenced this issue on Dec 15, 2020

`fix(fs): check resolved path against root` ··· ✗ `c80f0fd`

matt-forster mentioned this issue on Dec 15, 2020

fix(fs): check resolved path against root #224

Merged

matt-forster added a commit that referenced this issue on Dec 15, 2020

`fix(fs): check resolved path against root` ··· ✓ `b5d8fc0`

This comment has been minimized. Sign in to view

matt-forster added **security** and removed **help wanted** **input needed** labels on Dec 15, 2020

matt-forster closed this as completed in #224 on Dec 16, 2020

matt-forster pushed a commit that referenced this issue on Dec 16, 2020

`fix(fs): check resolved path against root (#224)` ··· ✓ `457b859`

matt-forster unpinned this issue on Dec 16, 2020

botovance commented on Dec 16, 2020 Contributor

🎉 This issue has been resolved in version 4.4.0 🎉

The release is available on:

- npm package (@latest dist-tag)
- GitHub release

Your **semantic-release** bot 📦 🚀

botovance added the **released** label on Dec 16, 2020

n-timofeev commented on Dec 16, 2020

Still not fixed or i'am doing something wrong...
demo1, demo2

👍 1

matt-forster commented on Dec 16, 2020 • edited Contributor

~~Hey @n-timofeev, could you confirm the version in the lock file you have in those videos?~~

I see you step through the new resolve in the second one, enough proof for me, apologies.

matt-forster reopened this on Dec 16, 2020

matt-forster added a commit that referenced this issue on Dec 16, 2020

`test: reproduce failing upper path test` … ✓ `1cc69a6`

matt-forster mentioned this issue on Dec 16, 2020

test: windows #232

Closed

matt-forster changed the title ~~User can escape from root directory~~ **Winddows: User can escape from root directory** on Dec 16, 2020

matt-forster changed the title ~~Winddows: User can escape from root directory~~ **Windows: User can escape from root directory** on Dec 16, 2020

matt-forster pinned this issue on Dec 16, 2020

matt-forster commented on Dec 16, 2020 • edited ▾ Contributor

I've got PR #232 setup to test windows in our CI environment - I've also added a test that uses the path as is in your demo videos.

There are a lot of tests failing in that branch right now, mostly due to tests expecting Unix output. This should get us on the right track to fixing this without a doubt.

matt-forster removed the **released** label on Dec 17, 2020

matt-forster commented on Dec 17, 2020 Contributor

GHSA-pmw4-jgxx-pcq9

n-timofeev commented on Dec 17, 2020 • edited ▾

Source of backslash
Server sends it in response to the PWD command and then Firefox prepend it in CWD

And i think, we need to block paths with sequential dots and slashes here as a temporary solution

This comment has been minimized. Sign in to view

renovate bot mentioned this issue on Feb 10, 2021

chore(deps): update dependency ftp-srv to 4.4.0 [security] - autoclosed watchdogpolska/docker-images#353

Closed

1 task

matt-forster added the **windows** label on May 28, 2021

antonilol commented on Nov 30, 2021

@Heartz66 making symlinks in you ftp folder is you own choice and cannot be done on the client side

Assignees

No one assigned

Labels

bug security windows

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

fix(fs): check resolved path against root
QuorumDMS/ftp-srv

8 participants