

main

Go to file

Code

alexp-sssup	Bump NPM deps	5c8642b · 2 months ago
.circleci	ci: add havana host k...	3 months ago
.github/workflows	Update deploy.yml	5 months ago
assets	add Claude instructi...	4 months ago
dockerfiles	Dockerfiles: debian_...	2 years ago
docs	Review README to ...	9 months ago
documents	Add example image	8 months ago
examples	Added luajit interp li...	2 years ago
src	Restore Alpine link a...	3 months ago
xterm	Bump xterm version	last year
.gitignore	add a basic .gitignore	4 months ago
.npmrc	Basic npm setup for ...	9 months ago
LICENSE.txt	Add Apache 2.0 licen...	2 years ago
README.md	add Claude instructi...	4 months ago
config_github_term...	Make sure GH config...	9 months ago
config_public_alpin...	Bump alpine image t...	4 months ago
config_public_termi...	Svelte: Expose more ...	9 months ago
favicon.ico	New favicon: red tow...	3 years ago
login.html	Revert "Experimenta...	3 years ago
nginx.conf	added mime types	8 months ago
package-lock.json	Bump NPM deps	2 months ago
package.json	Claude: Support dow...	4 months ago
postcss.config.js	Implement side-bar ...	4 months ago
scrollbar.css	Scrollbar	3 years ago
serviceWorker.js	serviceworker.js: Adj...	2 years ago
svelte.config.js	Enable SSR	9 months ago
tailwind.config.js	Import tailwind to ac...	9 months ago
towerico	Move favicon to red ...	3 years ago
vite.config.js	Expose the documen...	8 months ago

About

Virtual Machine for the Web

webvm.io

vmcppreplwebassemblywasmxterm.jscheerpxwebvmtailscalecheerpx

ReadmeApache-2.0 licenseActivityCustom properties14.5k stars86 watching2.6k forksReport repository

Contributors



+ 3 contributors

Languages

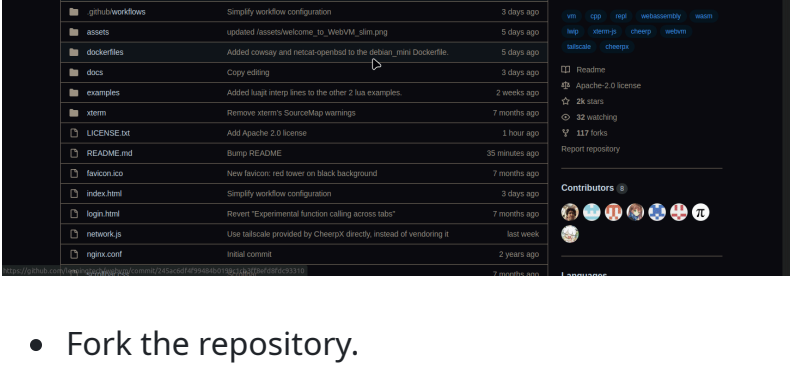
JavaScript 43.9%CSS 9.1%HTML 3.5%

WebVM

chat145 onlineissues11 open

This repository hosts the source code for <https://webvm.io>, a Linux virtual machine that runs in your browser.

Try out the new Alpine / Xorg / i3 graphical environment: <https://webvm.io/alpine.html>



WebVM is a server-less virtual environment running fully client-side in HTML5/WebAssembly. It's designed to be Linux ABI-compatible. It runs an unmodified Debian distribution including many native development toolchains.

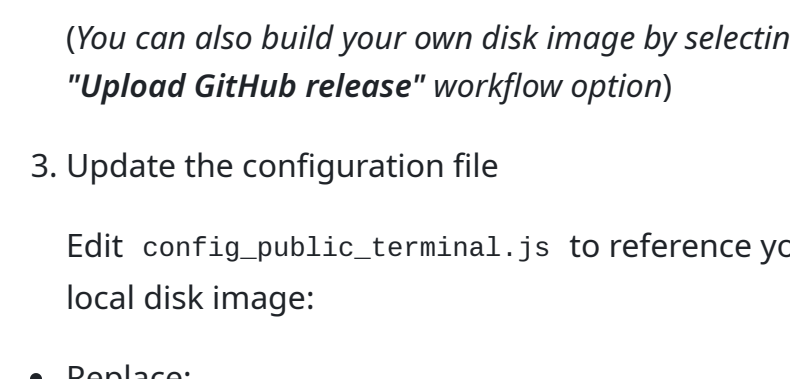
WebVM is powered by the CheerPX virtualization engine, and enables safe, sandboxed client-side execution of x86 binaries on any browser. CheerPX includes an x86-to-WebAssembly JIT compiler, a virtual block-based file system, and a Linux syscall emulator.

Enable networking

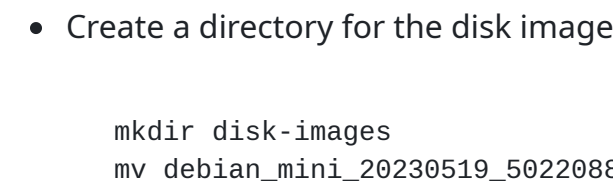
Modern browsers do not provide APIs to directly use TCP or UDP. WebVM provides networking support by integrating with Tailscale, a VPN network that supports WebSockets as a transport layer.

- Open the "Networking" panel from the side-bar
- Click "Connect to Tailscale" from the panel
- Log in to Tailscale (create an account if you don't have one)
- Click "Connect" when prompted by Tailscale
- If you are unfamiliar with Tailscale or would like additional information see [WebVM and Tailscale](#).

Fork, deploy, customize



- Fork the repository.
- Enable Github pages in settings.
 - Click on Settings .
 - Go to the Pages section.
 - Select Github Actions as the source.
 - If you are using a custom domain, ensure Enforce HTTPS is enabled.
- Run the workflow.
 - Click on Actions .
 - Accept the prompt. This is required only once to enable Actions for your fork.
 - Click on the workflow named Deploy .
 - Click Run workflow and then once more Run workflow in the menu.
- After a few seconds a new Deploy workflow will start, click on it to see details.
- After the workflow completes, which takes a few minutes, it will show the URL below the deploy_to_github_pages job.



You can now customize dockerfiles/debian_mini to suit your needs, or make a new Dockerfile from scratch. Use the Path to Dockerfile workflow parameter to select it.

Run WebVM locally with a custom Debian mini disk image

- Clone the WebVM Repository

```
git clone https://github.com/leaningtech/webvm
cd webvm
```
- Download the Debian mini Ext2 image

Run the following command to download the Debian mini Ext2 image:

```
wget "https://github.com/leaningtech/webvm/releases/download/v1.0.0/debian_mini_ext2.img"
```

(You can also build your own disk image by selecting the "Upload GitHub release" workflow option)
- Update the configuration file

Edit config_public_terminal.js to reference your local disk image:

 - Replace:

```
"wss://disks.webvm.io/debian_large_20230522_5044875331.ext2"
```

With:

```
"/disk-images/debian_mini_20230519_5022088024.ext2"
```

(Use an absolute or relative URL pointing to the disk image location.)

 - Replace "cloud" with the correct disk image type: "bytes"
- Build WebVM

Run the following commands to install dependencies and build WebVM:

```
npm install
npm run build
```

The output will be placed in the build directory.

- Configure Nginx
 - Create a directory for the disk image:

```
mkdir disk-images
mv debian_mini_20230519_5022088024.ext2 d:
```

 - Modify your nginx.conf file to serve the disk image. Add the following location block:

```
location /disk-images/ {
    root .;
    autoindex on;
}
```

- Start Nginx

Run the following command to start Nginx:

```
nginx -p . -c nginx.conf
```

Nginx will automatically serve the build directory.
- Access WebVM

Open a browser and visit: http://127.0.0.1:8081 .

Enjoy your local WebVM!

Example customization: Python3 REPL

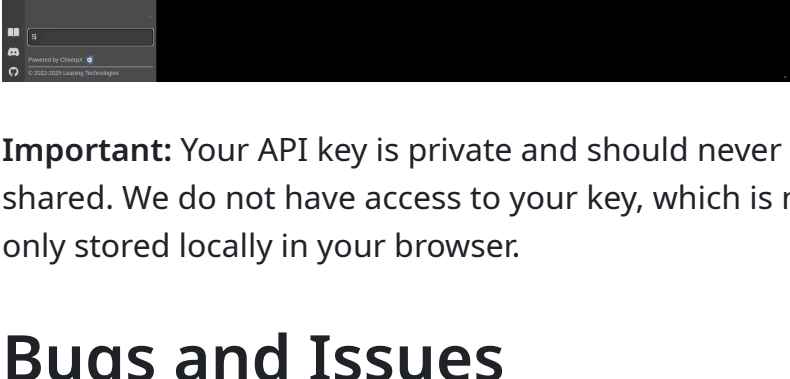
The Deploy workflow takes into account the CMD specified in the Dockerfile. To build a REPL you can simply apply this patch and deploy.

```
diff --git a/dockerfiles/debian_mini b/dockerfiles/debian_mini
index 2878332..1f3103a 100644
--- a/dockerfiles/debian_mini
+++ b/dockerfiles/debian_mini
@@ -15,4 +15,4 @@ WORKDIR /home/user/
# We set env, as this gets extracted by webvm
ENV HOME="/home/user" TERM="xterm" USER="user"
RUN echo 'root:password' | chpasswd
-CMD [ "/bin/bash" ]
+CMD [ "/usr/bin/python3" ]
```

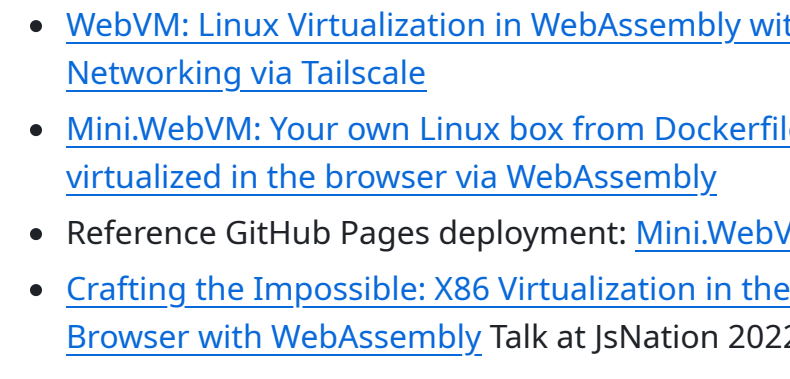
How to use Claude AI

To access Claude AI, you need an API key. Follow these steps to get started:

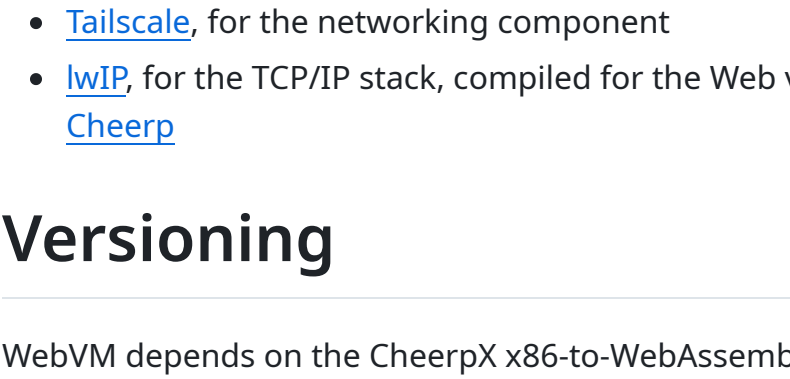
- Create an account
 - Visit [Anthropic Console](#) and sign up with your e-mail. You'll receive a sign in link to the Anthropic Console.



- Get your API key
 - Once logged in, navigate to **Get API keys**.
 - Purchase the amount of credits you need. After completing the purchase, you'll be able to generate the key through the API console.



- Log in with your API key
 - Navigate to your WebVM and hover over the robot icon. This will show the Claude AI Integration tab. For added convenience, you can click the pin button in the top right corner to keep the tab in place.
 - You'll see a prompt where you can insert your Claude API key.
 - Insert your key and press enter.



Important: Your API key is private and should never be shared. We do not have access to your key, which is not only stored locally in your browser.

Bugs and Issues

Please use [Issues](#) to report any bug. Or come to say hello / share your feedback on [Discord](#).

More links

- [WebVM: server-less x86 virtual machines in the browser](#)
- [WebVM: Linux Virtualization in WebAssembly with Full Networking via Tailscale](#)
- [Mini.WebVM: Your own Linux box from Dockerfile, virtualized in the browser via WebAssembly](#)
- Reference GitHub Pages deployment: [Mini.WebVM](#)
- [Crafting the Impossible: X86 Virtualization in the Browser with WebAssembly](#) Talk at JsNation 2022

Thanks to...

This project depends on:

- [CheerPX](#), made by [Leaning Technologies](#) for x86 virtualization and Linux emulation
- [xterm.js](#), <https://xtermjs.org/>, for providing the Web-based terminal emulator
- [Tailscale](#), for the networking component
- [lwip](#), for the TCP/IP stack, compiled for the Web via [Cheerpx](#)

Versioning

WebVM depends on the CheerPX x86-to-WebAssembly virtualization technology, which is included in the project via [NPM](#).

The NPM package is updated on every release.

Every build is immutable, if a specific version works well for you today, it will keep working forever.

License

WebVM is released under the Apache License, Version 2.0.

You are welcome to use, modify, and redistribute the contents of this repository.

The public CheerPX deployment is provided **as-is** and is **free to use** for technological exploration, testing and use by individuals. Any other use by organizations, including non-profit, academia and the public sector, requires a license. Downloading a CheerPX build for the purpose of hosting it elsewhere is not permitted without a commercial license.

Read more about [CheerPX licensing](#)

If you want to build a product on top of CheerPX/WebVM, please get in touch: sales@leaningtech.com

