

ArchiveBox / archivebox-browser-extension Public

- <> Code
- Issues 9
- Pull requests
- Actions
- Wiki
- Security and quality
- Insig

6 Branches
 3 Tags

pirate	Fix extension sync and packaging flow	f262471 · last week	
docs	Upgrade extension UI and loc...	last month	
entrypoints	Fix extension sync and packa...	last week	
locales	add translations, playwright te...	last month	
public/icon	Upgrade extension UI and loc...	last month	
src	Fix extension sync and packa...	last week	
tests	Fix extension sync and packa...	last week	
.env.example	Add Edge and Safari browser ...	last month	
.gitignore	Ignore Playwright extension b...	last month	
LICENSE	fix icons	last year	
PRIVACY.md	Add browser extension privac...	last month	
README.md	Update links and descriptions ...	last month	
package.json	Fix extension sync and packa...	last week	
playwright.config.ts	add translations, playwright te...	last month	
pnpm-lock.yaml	add translations, playwright te...	last month	
tsconfig.json	Upgrade extension to WXT R...	last month	
wxt.config.ts	Fix extension sync and packa...	last week	





README
 MIT license

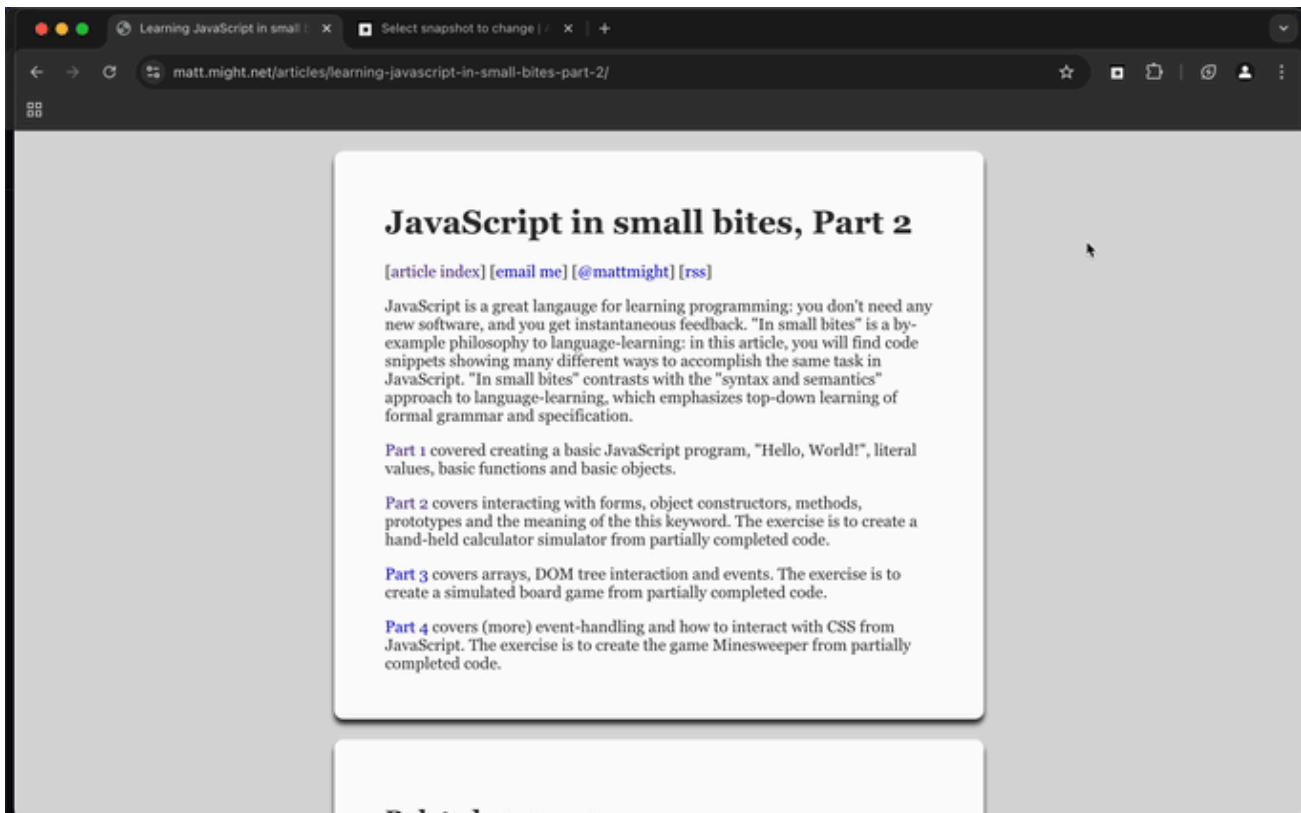
ArchiveBox Browser Extension

This is a browser extension that lets you send individual browser tabs or all URLs matching certain patterns to your [ArchiveBox](#) instance for offline preservation. This has a couple of benefits:

- Own your data: save the web content that matters to you most, protect against link rot
- Protect your data: save offline copies of pages in common, durable formats that will last for generations
- Use your data: collect and tag important bookmarks, full-text search through your browsing history, automatically push captured data into other systems using ArchiveBox's APIs

Get the Extension

-  [Chrome / Brave / Other Chromium-based browsers](#)
-  [Firefox / Waterfox / Tor Browser / Other Firefox-based browsers](#)
-  [Microsoft Edge](#)
-  Safari / iOS Safari (*supported; requires manual install*)



The screenshot shows the ArchiveBox web interface. At the top, there's a navigation bar with 'ArchiveBox' logo and links for 'SNAPSHOTS', 'TAGS', 'LOG', 'DOCS', 'API', 'PUBLIC', 'ADMIN', 'USER ARCHIVEBOX', and 'ACCOUNT / LOG OUT'. Below the navigation bar is a search bar with '59 results (Show all)'. There are also buttons for 'Edit tags', '+', '-', 'Get Title', 'Get Missing', 'Archive Again', 'Redo', and 'Delete'. A table of archived items is displayed with columns: 'CREATED AT', 'TITLE', 'FILES SAVED', 'SIZE', and 'ORIGINAL URL'. The table lists various archived pages, including 'Matt Might', 'Some terminal frustrations', 'JavaScript's Quirky eval', 'Another Perl One-liner: Byte Order', and several 'Day' series from a blog. A 'FILTER' sidebar on the right offers options to filter by 'created at', 'downloaded at', 'status', and 'created by'.

CREATED AT	TITLE	FILES SAVED	SIZE	ORIGINAL URL
2025-02-13 05:49:35 PM	Matt Might		...	https://matt.might.net/
2025-02-13 01:52:34 PM	★ Some terminal frustrations		266.8 KB	https://jvns.ca/blog/2025/02/05/some-terminal-frustrations/
2025-02-13 01:40:30 PM	🟢 JavaScript's Quirky eval		260.4 KB	https://nullprogram.com/blog/2012/11/14/
2025-02-13 12:59:53 PM	🟢 Another Perl One-liner: Byte Order		248.6 KB	https://nullprogram.com/blog/2009/05/20/
2025-02-13 12:29:43 PM	★ Day 6: I wrote a rootkit!		251.6 KB	https://jvns.ca/blog/2013/10/08/day-6-i-wrote-a-rootkit/
2025-02-13 12:28:00 PM	★ Day 5: I wrote a kernel module!!!		252.2 KB	https://jvns.ca/blog/2013/10/07/day-5-i-wrote-a-kernel-module/
2025-02-13 10:10:38 AM	★ Where to find bike sharing systems' data feeds		253.1 KB	https://jvns.ca/blog/2013/10/06/where-to-find-the-bixi-xml-data-feed/
2025-02-13 10:08:20 AM	★ Day 4: Processes vs threads, and kernel modules!		252.0 KB	https://jvns.ca/blog/2013/10/04/day-4-processes-vs-threads/
2025-02-13 10:07:09 AM	★ Day 3: What does the Linux kernel even do?		251.2 KB	https://jvns.ca/blog/2013/10/02/day-3-what-does-the-linux-kernel-even-do/
2025-02-13 09:42:54 AM	★ Day 2: netcat fun!		249.4 KB	https://jvns.ca/blog/2013/10/01/day-2-netcat-fun/
2025-02-13 09:42:54 AM	★ Day 1: What does a		...	https://jvns.ca/blog/2013/09/30/hacker-

URLs Config **Personas**

ArchiveBox Server URL *

https://demo.archivebox.io

ADMIN TEST SERVER

The base URL of your self-hosted archivebox server, e.g. http://localhost:8000 200 (50ms)

ArchiveBox API Key

70e957dbd1d9fd8d730b66a85fc3faa1

+ GENERATE TEST API KEY

ArchiveBox API key generated on your server (32 character lowercase hex string) API key valid (user 1) (36ms)

Match URL Patterns

*

Regex of URLs to archive automatically whenever they are visited

Exclude URL Patterns

:/mail.google.com

Regex of URLs to never archive automatically

Test URL Against Patterns

https://example.com

TEST ADDING

URL would be allowed to be archived Adding failed: Error: Request failed 401

URLs Config **Personas** Import

Active Persona: Nickbot

Create New Persona

Nickbot: 4 domains, 122 cookies total

Managed Personas

Detect Current Settings

Name	Domains	Last Used	Settings	Actions	
Nickbot	4 domains	11/27/2024, 3:27:27 AM	User Agent Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML)	Geography Berkeley, United States Timezone America/Los_Angeles Language en-US Operating System macOS Viewport Size 1866x1094	Export Cookies Delete

Import Browser Cookies

Import cookies from domains you're currently logged into to the active persona.

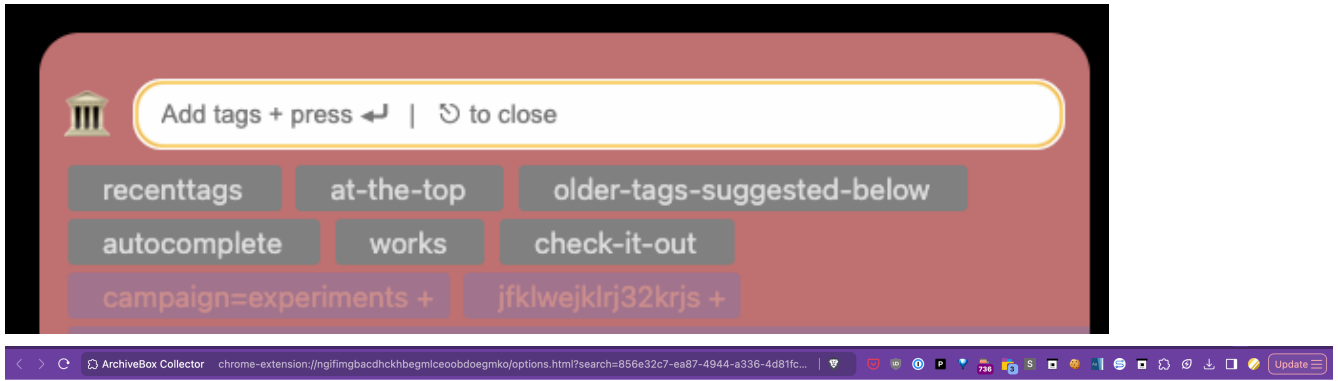
Filter github

Select All Deselect All

Domain	Cookies	Preview
<input checked="" type="checkbox"/> github.com	12	Preview
<input type="checkbox"/> gist.github.com	3	Preview

Selected: 1 Select All Deselect All

Import Cookies to Active Persona



ArchiveBox Saved URLs

Filter entries...

2024-11-26 06:04:52	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	github test-tag tag=123 depth=1 SAVE_WARC=False
2024-11-26 06:05:12	https://github.com/GoogleChrome/chrome-extensions-samples/blob/main/api-samples/web-a-	
2024-11-26 06:05:37	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	
2024-11-26 06:05:38	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	
2024-11-26 06:05:38	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	
2024-11-26 06:05:39	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	
2024-11-26 06:05:44	https://stackoverflow.com/questions/26140443/executing-code-at-page-level-from-backgr-	
2024-11-26 06:06:04	https://stackoverflow.com/questions/32523274/chrome-extension-callback-in-background-j-	github tag=123 test-tag

All Tags

github	2
test-tag	2
tag=123	2
depth=1	1
SAVE_WARC=False	1

URLs Config Personas Import

Chrome History Chrome Bookmarks

From 2024-11-26 To 2024-11-27 Load History

1865 Show new only Selected: 1 Select All Deselect All

<input type="checkbox"/>	URL	Title	Timestamp
<input checked="" type="checkbox"/>	chrome-extension://noogafoofpebimajpfpamcfoaifemoa/suspended.html#ttl=Marketplace%20-%20Asus%20Prime%20Z790%20-%20A%20Wifi%20Motherboard%20%7C%20Facebook&pos=0&uri=https://www.facebook.com/marketplace/item/1118758186528627/?ref=search&referral_code=null&referral_story_type=post&tracking=browse_serp%3Aea21773b-d1b7-4e26-8bc5-7565b4296877	Marketplace - Asus Prime Z790 - A Wifi Motherboard Facebook	11/26/2024, 6:38:42 PM

Tags tag1,tag2 Import Selected

Recent Changes

- ✓ updated the extension to Manifest v3 using WXT, React, and TypeScript
- ✓ added a Saved URLs view where you can see, search, sort, tag, sync, delete, and export the URLs you've collected so far
- ✓ added the ability to import URLs from browser history / bookmarks by date range or filter query
- ✓ added the ability to export selected URLs as CSV/JSON, selected screenshots as PNG, selected MHTML snapshots as `.mhtml`, selected SingleFile captures as `.html`, or a ZIP

bundle containing all selected snapshot data and local artifacts

- ✓ added extension-local full-page screenshot capture for saved URLs
- ✓ added extension-local MHTML capture for saved URLs on Chrome / Edge / Chromium browsers
- ✓ added extension-local SingleFile HTML capture for saved URLs when the SingleFile extension is installed and approved
- ✓ added the ability to edit extension config options, allowlist/denylist, ArchiveBox server URL, API key, and authentication profiles from the options page
- ✓ added the ability to test the connection to your ArchiveBox server
- ✓ added build and packaging support for Chrome, Edge, Firefox, and Safari from the WXT/React codebase

Local Captures

When local capture saving is enabled in the options page, the extension stores capture artifacts in the browser's extension-local OPFS storage before the popup is shown:

- Full-page screenshot:

```
snapshots/YYYYMMDD/example.com/{uuid}/chrome_extension_screenshot/screenshot.png
```

- MHTML snapshot:

```
snapshots/YYYYMMDD/example.com/{uuid}/chrome_mhtml/snapshot.mhtml
```

- SingleFile HTML snapshot:

```
snapshots/YYYYMMDD/example.com/{uuid}/chrome_extension_singlefile/snapshot.html
```

Screenshots are shown as thumbnails in the Saved URLs table when at least one saved URL has a screenshot, and can be exported as PNG from the Export menu. MHTML and SingleFile HTML snapshots can also be exported from the same menu. The ZIP export includes the selected CSV/JSON metadata plus local artifacts under the same

`snapshots/YYYYMMDD/example.com/{uuid}/...` paths used in OPFS. When a SingleFile HTML or MHTML snapshot is available, the Saved URLs table title opens an extension-local framed viewer for that snapshot.

MHTML capture uses Chromium's `pageCapture.saveAsMHTML()` extension API and is available in Chrome / Edge / Chromium builds. Firefox and Safari builds still save the URL and screenshot where the browser supports tab capture, but skip MHTML capture because the page capture API is not available there.

SingleFile HTML capture uses the SingleFile browser extension through its external capture API. The first request opens SingleFile's options page so the user can approve or deny ArchiveBox as an allowed caller, then subsequent captures can run silently and save the returned HTML into ArchiveBox's local OPFS snapshot tree.

Setup

1. Set up an [ArchiveBox](#) server and make sure it's accessible to the machine you're browsing on
2. Configure your ArchiveBox server to allow URL submissions without requiring login ([more info here...](#))

`>= v0.8.5` : users of the new BETA releases can use an API key generated at `/admin/api/apitoken/` instead.

Alternatively: if you stay signed in to your ArchiveBox instance in the same browser, it will share your login credentials.

```
archivebox config --set PUBLIC_ADD_VIEW=True
# (make sure to restart the server after if you apply this change)
```



```
~/o/A/data / (main) *1 # archivebox config --set PUBLIC_ADD_VIEW=True
[i] [2024-01-23 19:11:05] ArchiveBox v0.7.2: archivebox config --set PUBLIC_ADD_VIEW=True
> /Users/squash/opt/ArchiveBox/data

PUBLIC_ADD_VIEW=True
~/o/A/data / (main) *1 # archivebox server --debug
[i] [2024-01-23 19:11:16] ArchiveBox v0.7.2: archivebox server --debug
> /Users/squash/opt/ArchiveBox/data

[+] Starting ArchiveBox webservice...
> Logging errors to ./logs/errors.log
Performing system checks...

System check identified no issues (0 silenced).
January 23, 2024 - 19:11:17
Django version 3.1.14, using settings 'core.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[+] Adding URL: https://example.com/

[+] [2024-01-23 19:11:21] Archiving 1/182 URLs from added set...
```

3. Configure the extension to point to your ArchiveBox server's base URL (e.g. `http://localhost:8000` , `https://archivebox.example.com` , etc.)

The top screenshot shows a browser window with the URL `nullprogram.com/blog/2012/11/19/`. The page title is "JavaScript Debugging Challenge" by Chris Welons, dated November 19, 2012. The article text reads: "As I've been exploring JavaScript I've come up with a few interesting questions that expose some of JavaScript's quirks. This is the first one, which I came up with over the weekend. Study it and try to come up with an answer before looking at the explanation below. Go ahead and use a JavaScript interpreter or debugger to poke at it if you need to." Below the text is a code block containing JavaScript code for a function that builds a table and fills it with numbers. The code is as follows:

```
var count = 4;

function foo() {
  var table = [count];

  /* Build the table. */
  while (count-- > 0) {
    table.push([]);
  }

  /* Fill it with numbers. */
  for (var count = 1; count < table.length; count++) {
    table[count].push(count);
  }
  return table;
}

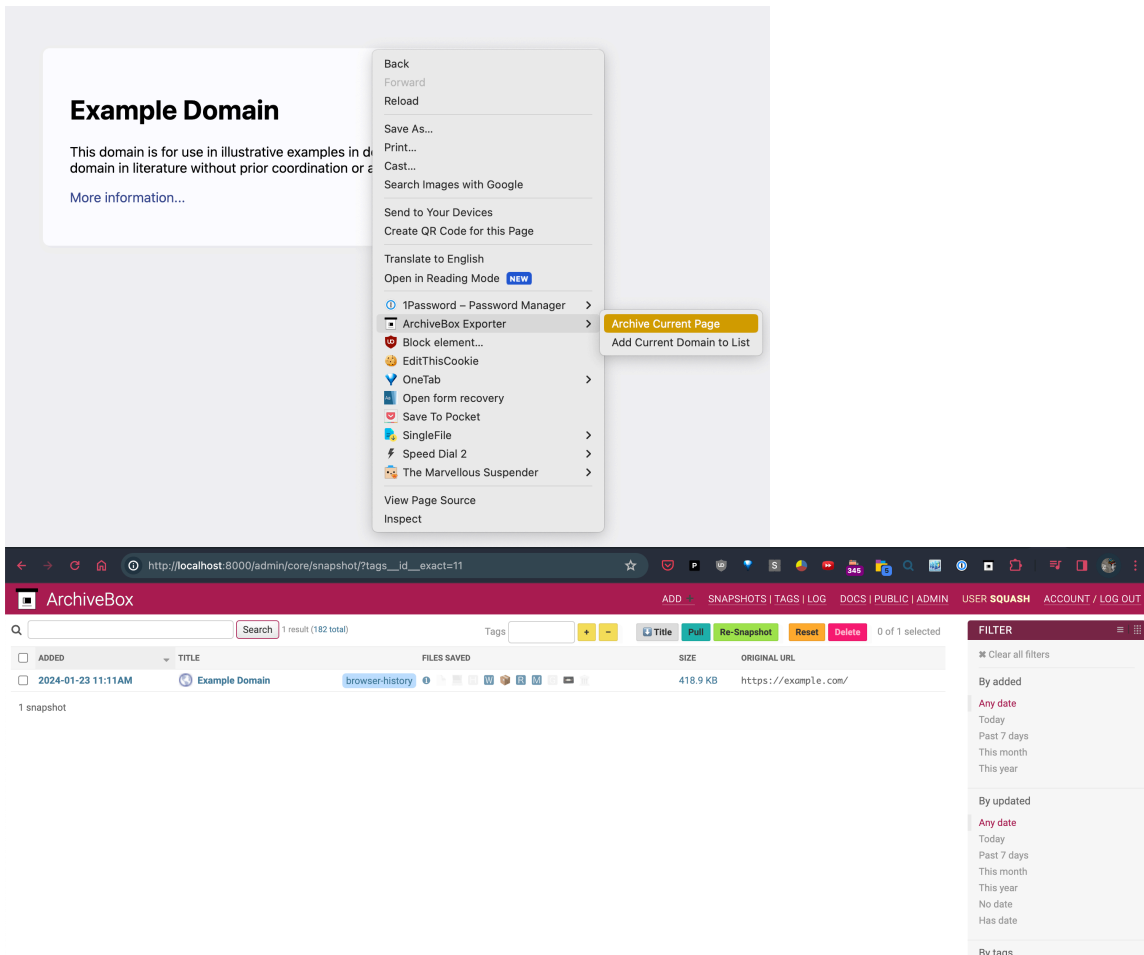
foo(); // What does this return? And why?
```

Below the code, the author notes: "When I originally came up with the problem, I just enabled [impatient-mode](#) in my editor buffer to share it friends. It's a really convenient alternative to `nastebins!`"

The bottom screenshot shows the configuration window of the ArchiveBox browser extension. It includes the following settings:

- Config** (with a [Need help?](#) link)
- Archive Mode**: A dropdown menu set to "Allowlist (don't archive by default)".
- ArchiveBox Base URL**: A text input field containing `https://demo.archivebox.io`.
- Tags**: A text input field containing `browser-history`.
- Archived Domains**: A section with the text "Pages you visit on these domains will be sent to your ArchiveBox. Wildcard subdomains are allowed, e.g. *.google.com". It features a "Regex?" checkbox, a "Domain to add" input field, and "+", "-", and "Clear" buttons. Below this, a list of domains is shown: `en.wikipedia.org/.*` and `youtube.com/.*`, with the text "Regex entries are listed in blue. You can find more docs [here](#)."

4. Test it out by right-clicking on any page and selecting `Save to ArchiveBox`, or by clicking the extension icon in the menubar.



Development

🌟 Originally contributed by [TJ Horner \(@tjhorner\)](#), now maintained by [@benmuth](#) and the [ArchiveBox](#) team.

If you wish to contribute to (or just build for yourself) this extension, you will need to download and install [Node.js](#) and [pnpm](#).

```
git clone https://github.com/ArchiveBox/archivebox-browser-extension
cd archivebox-browser-extension/
```

```
pnpm install
pnpm compile
pnpm build
```

Contributors 7



Languages

