

1 We Have Serverless at Home

By: Ryan Hornby and Matthew Safar

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0,
      user-scalable=no">
    <title>Serverless At Home: Sigbovik 2024</title>
    <link rel="icon" type="image/x-icon" href="favicon.png">
    <link rel="stylesheet" href="https://cdn.rawgit.com/Chalarangelo/mini.css/v3.0.1/dist/
      mini-default.min.css">
    <link rel="stylesheet" href="style.css">
    <script defer src="./script.js" type="module"></script>
  </head>
  <body>
    <header class="sticky">
      <a href="https://sigbovik.org/" class="logo"></a>
      <div id="header-contents">
        <div id="header-top">
          Sigbovik 2024
        </div>
        <div id="header-bottom">
          <a href="/" class="button" style="padding-left: 0;">Home</a>
          <a href="ringfuck.html" class="button">Ringfuck</a>
          <a href="404.html" class="button">404</a>
        </div>
      </div>
    </header>
    <header class="sticky ruler-container">
      <div class="ruler ruler-x"></div>
    </header>
    <div class="ruler-container" id="y-ruler-bar">
      <div class="ruler ruler-y"></div>
    </div>
    <div id="paper">
      <h1 class="title">Serverless at Home?</h1>
      <h4 class="title">Ryan Hornby and Matthew Safar</h4>
      <h4><b>Abstract:</b></h4>
      </img>
      <h2>Introduction</h2>
      <p>The client-server model has been in use for around 60 years, but over the
        past two decades, the idea of serverless computing has become feasible and
        popular. In case you aren't familiar with serverless computing, allow the
        collective consciousness of StackExchange (ChatGPT) to explain: "Serverless
        computing? More like 'server-less headaches.' It's not truly 'serverless'
        because, surprise, servers still exist! You're just off the hook for
        babysitting them. Think of it more as 'servers on demand' with a fancy
        billing twist." While not babysitting servers sounds great, most people
        would prefer not getting billed for the pleasure. In this paper, we will
        investigate "free serverless", also known as serverless at home. </p>
      <h2>Methods</h2>
      <h3>Hosting</h3>
      <p>For hosting, we use Github Pages. 'Its a free, powerful solution for your
        frontend AND backend needs! You can create your stunning frontend with HTML
        /CSS/Javascript and power it with Github Actions to produce a functional
        and modern website! </p>
```

<h3>WASM</h3>

<p>Perhaps a more intuitive definition of serverless would be one where there simply is no server (outside of the server hosting the html files). At first, this definition may sound too restrictive to have interesting websites, but by using web assembly (WASM) and javascript we can make the client run all the `code` to make our site interesting. We've shown this functionality with our previous SIGBOVIK entry: Ringfuck. While WASM makes creating interesting websites pretty easy, javascript psychopaths have been doing the same thing `for` years without it.</p>

<h3>CI/CD</h3>

<p>If we want true serverless at home, we need to be able to replicate the functionality of database-driven websites `for` free. To do this, we take inspiration from others who have sought to innovate by taking other 'peoples backend and making it their own (see https://www.youtube.com/watch?v=c_arQ-6ElYI). For our backend we leverage Github and its most recent feature (besides stealing everyone's `code` to train a Kamikaze CoPilot) CI/CD, or Github Actions as they call it. Github's CI/CD allows you to write a custom bash script to be run in a Docker container. This allows us to edit our html with simple CLI tools like cat, echo, sed, and awk and then commit those changes back to the repo. The final step is to simply trigger the CI/CD to run when a user performs an `action` requiring a server operation. To perform this, we use some simple javascript along with a Github auth token to trigger the `action`, then query the Github API to determine when the change has finished so that we can automatically refresh the page.</p>

<p>Something to consider is that our choice to stick to CLI tools is only because we wanted to stay simple and fast. If those are not concerns `for` your website, you can easily set up a CI/CD pipeline that includes advanced tools like Python `for` machine learning, SQLite `for` relational databases, or Elasticsearch `for` search engine capabilities! The possibilities are truly endless, as long as Github 'doesnt notice your compute time and start charging you money. </p>

<h2>Security and Privacy</h2>

<p>While there are security concerns with this serverless at home methodology, a lot of potential security issues are shared with other applications that accept user input, so we will not be discussing these. A unique issue with our CI/CD approach is the need `for` a publicly accessible Github auth token. To mitigate this risk, it is recommended to use a fine-grained access token `for` the specific repository. The major downside to this is that the token must have an expiration date, thus creating some maintenance. To avoid maintenance, you could create a dummy account with a classic access token, but this would allow anyone to attempt changes on public repos with your dummy account.</p>

<p>On the privacy front, there are a couple of unique issues with serverless at home. First is the classic git problem of it being very difficult to permanently delete `content` that was committed. Thus, any sensitive `data` in a comment will remain visible to everyone until an administrator removes it, which may never happen since owners of github repos are notoriously picky about other people changing their precious `code`. The next concern regards user sessions. While it is possible to have the CI/CD system create user sessions by creating subdirectories and deleting them later, these sessions would be viewable by anyone interested enough to pull up the source `code`. Since this is not easily mitigatable, our recommendation is to simply inform the users of your site that they have no privacy. So if you've read this far, congratulations you've been informed.</p>

<h2>Comparison to Traditional Serverless</h2>

<p>Traditional hosting and serverless are a pretty polished experience `for` devs and users, especially compared to serverless at home. So why would you ever use serverless at home? One selling point is the fact that it is

totally free, as long as you don't have a bunch of users, which shouldn't be a big concern for most readers. Another benefit of serverless at home is the nostalgia for dial-up internet. Some people would argue there are better free alternatives, like Wordpress. Which brings us to our next selling point: freedom. Services like Wordpress might be free, but they restrict what you can do with your site to try to get you to pay for more freedom. But with serverless at home, as long as you can code it to run when the Github Action triggers, you're free to do it. Finally, the biggest reason to use serverless at home is so that you can deliver an awful user experience to your "friend" when they ask you to make a "simple" website for them.

Conclusion

Is serverless at home revolutionary? Let us know in the comments! 🙌

<dialog id="loading-dialog">

<div class = "card large">

<p>Please wait while we post your comment ! 🤖</p>

<audio id="loading-sound" loop preload>

<source src="/the-sound-of-dial-up-internet-6240.mp3" type="audio/mpeg" />

</audio>

</div>

</dialog>

<div id="comments">

<h3>Comments</h3>

<form id="comment-form">

<fieldset>

<div class="row responsive-label"><div class="col-sm-12 col-md-3"><label for="cf-name">Name</label></div>

<div class="col-sm-12 col-md"><input type="text" placeholder="Name" id="cf-name" name="Name" style="width:85%;"></div></div>

<div class="row responsive-label"><div class="col-sm-12 col-md-3"><label for="cf-textarea">Textarea</label></div>

<div class="col-sm-12 col-md"><textarea style="width:85%;" placeholder="Textarea" id="cf-textarea" name="Textarea"></textarea></div></div>

<button>Post Comment</button>

</fieldset>

</form>

<pre>Ryan Hornby: First</pre>

</div>

</div>

</body>

</html>

2 Appendix 1

List of relevant links:

1. <https://backendathome.github.io/>
2. <https://backendathome.github.io/serverless.png>
3. <https://backendathome.github.io/ringfuck.html>
4. https://www.youtube.com/watch?v=c_arQ-6ElYI

3 Appendix 2

Preview of site:

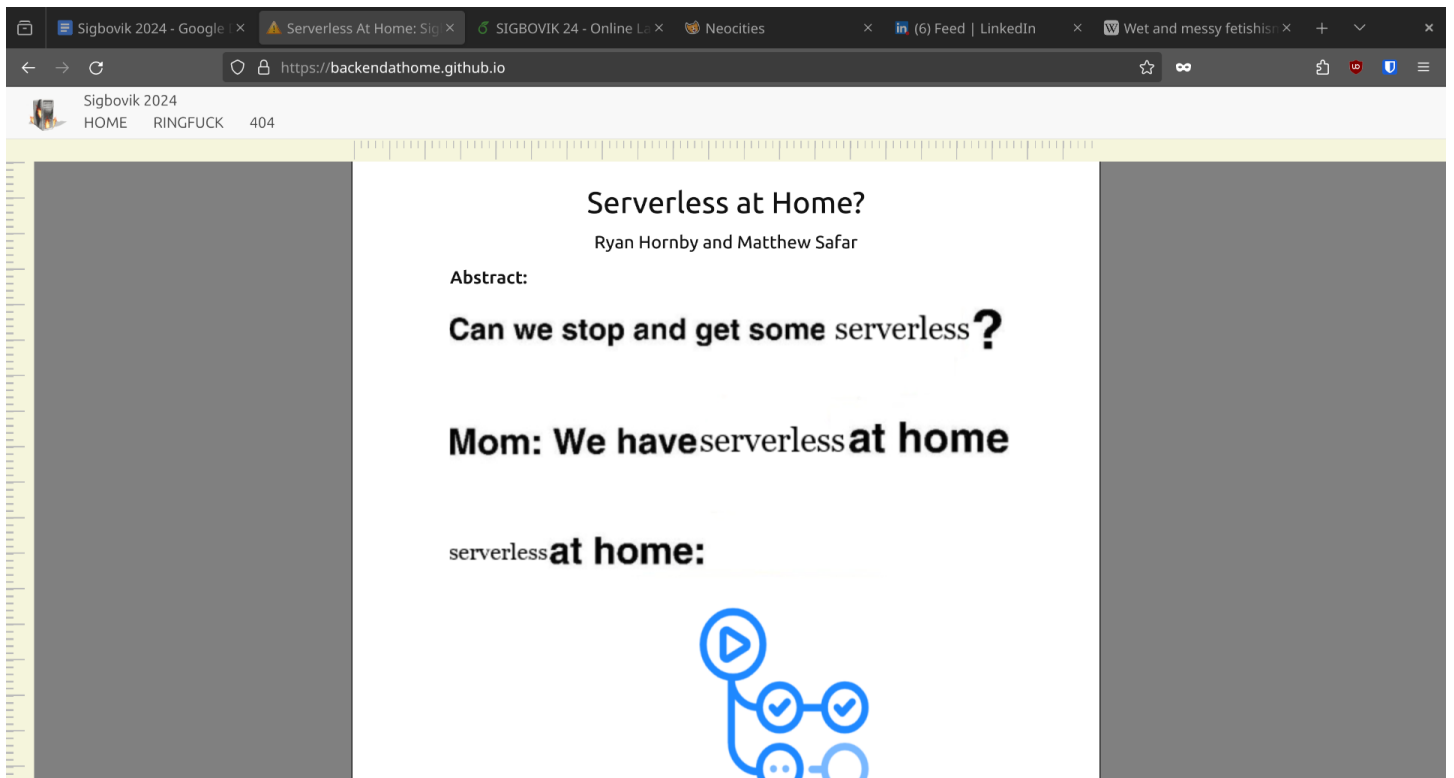


Figure 1: Example of serverless website.