

If you've ever accessed a website behind an Anubis anti-bot page, you might have noticed it showing something like **Speed: 0kH/s** that never updates. This can be confusing and annoying, especially if you want to get to the content quickly. In this post, we'll explain why these anti-bot challenges exist, what that confusing speed number means in simple terms, and how modern browsers and JavaScript play a key role in solving the "speed 0kH/s fix" mystery.

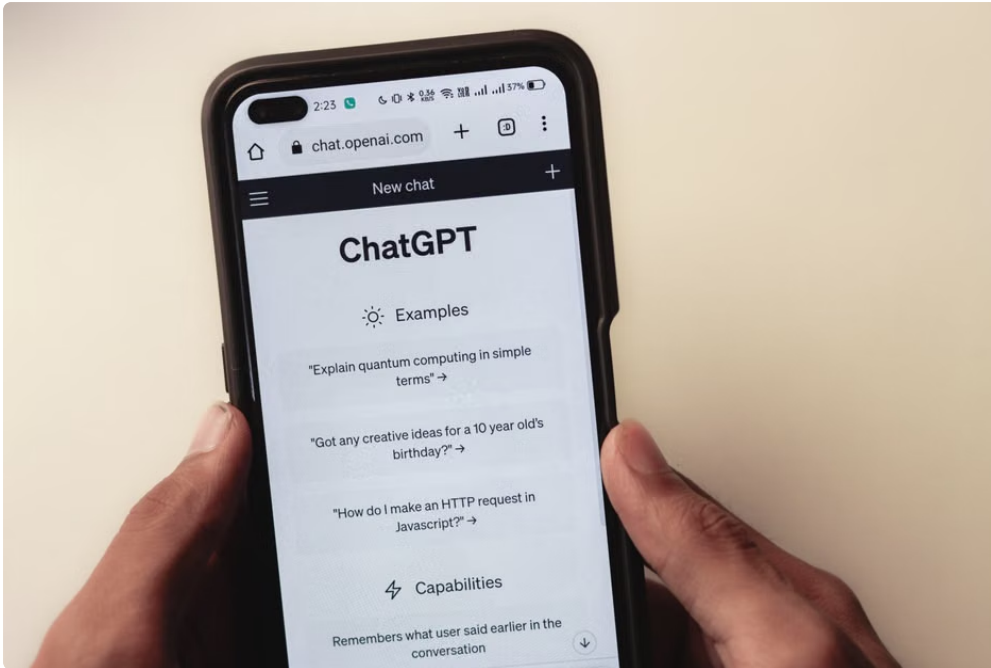


Table of Contents

1. Why Anti-Bot Pages Exist
2. Proof-of-Work in Plain English
3. Hashcash Background
4. JavaScript Requirements and Modern Features
5. Common JavaScript Blocked Symptoms and How to Fix

Why Anti-Bot Pages Exist

Before diving into the specifics of Anubis and that "speed 0kH/s" indicator, it's important to understand why anti-bot pages, like those presented by Anubis, exist in the first place.

- **Preventing Bots from Scraping or Overloading Sites:** Many websites want to protect their content, user data, and infrastructure from malicious bots that scrape data or send excessive automated requests which can cause downtime.
- **Reducing Spam and Fraud:** Bots are often used to submit fake forms, spam comments, or engage in fraudulent activities like credential stuffing. Anti-bot measures help keep these activities at bay.
- **Ensuring Fair Use and Performance:** When genuine users share bandwidth or server processing with aggressive bots, site speed and reliability suffer. Anti-bot challenges like Anubis help preserve good performance for humans.

Websites often use CAPTCHAs or other challenges for bot filtering. Anubis is a bit different—it uses a Proof-of-Work system to ask your browser to do some computational work before you get access. That helps identify whether you are a browser executing scripts or a simple bot/fake client.

Proof-of-Work in Plain English

Proof-of-Work (PoW) is a way for websites to ask your computer to solve a small puzzle before they let you in. Think of this as the computer equivalent of "showing your ticket" [boerse-social](#) before you enter a concert.

Here's the simple version:

1. The website sends a challenge that requires a little math problem to be solved.
2. Your computer (more specifically, your browser's JavaScript engine) tries to find a number that fits the website's criteria.
3. Once your browser finds the right number, it gets access.

This process requires your device to spend some CPU time calculating, which is why you might see a "speed" measurement like "0kH/s" on the Anubis screen — it's showing how fast your device is performing these computations (measured in kilo-hashes per second).

Why use Proof-of-Work? It slows down bots because they'd need a lot of computing power to solve these puzzles repeatedly and fast, which deters automated attacks, but it's still fast and easy enough for most human user devices to handle.

How does this look practically?

Imagine you're asked to find a four-digit code between 0000 and 9999 so that when combined with the website's challenge string and processed through a cryptographic hash (like SHA-256), the result starts with a few zeros. Your browser keeps trying different numbers until it finds the right one.

Hashcash Background

Anubis's Proof-of-Work is based on a concept called Hashcash, originally designed to reduce spam emails.

Term Explanation Hash Function A function like SHA-1 or SHA-256 that takes an input and returns a fixed-size string (hash). Even a tiny change in input drastically changes the hash. **Nonce** A number that your computer guesses and increments during solution attempts. **Difficulty** How hard it is to find a hash that meets specific requirements (e.g., starts with 4 zero bits).

Hashcash requires the sender (or browser in this case) to find a nonce resulting in a hash with a certain property (like leading zeros). It's computationally costly but easy to verify.

This system ensures only genuine browsers, which can execute JavaScript and perform calculations, proceed. Simple bots or scrapers that don't support JS or can't perform work fast will fail or get stuck.

JavaScript Requirements and Modern Features

Because Anubis relies on Proof-of-Work in the browser, JavaScript must be fully enabled, supported, and able to execute modern language features.

Here's why your browser's JavaScript is essential:

- **Calculations run client-side:** The device you are using (phone, laptop) must perform the work, so the browser needs to run scripts.
- **Modern JavaScript Features:** Many Proof-of-Work implementations rely on ES6+ features such as Promises, async/await, and Uint8Array for efficient processing.
- **Crypto APIs:** The `window.crypto.subtle.digest()` API is often used for fast SHA hashing natively supported by browsers.

Common reasons JavaScript might be blocked or limited:

- **Browser extensions:** Security or privacy add-ons can disable scripts or interfere with execution.
- **Strict corporate firewalls or proxy policies:** Sometimes these block or alter JavaScript code.
- **Outdated browsers or devices:** Older browsers may not support required modern JavaScript or crypto APIs.
- **Explicit browser settings:** Some users disable JavaScript manually or block specific features.

Without the ability to run JavaScript properly, your browser cannot complete the Proof-of-Work, so the Anubis page shows **Speed: 0kH/s** indefinitely.

Common JavaScript Blocked Symptoms and How to Fix

If you see the "Speed: 0kH/s" stuck on Anubis and never changing, here's a checklist of what might be wrong and the fastest fix to get through:

Quick troubleshooting checklist:

1. **Check JavaScript is Enabled:** Go to your browser settings and confirm JavaScript is allowed. This is almost always required.
2. **Disable Extensions Temporarily:** Ad blockers, script blockers, or privacy extensions can block or interfere with the execution of Proof-of-Work JS.
3. **Update Your Browser:** Make sure you are using a modern browser version (Chrome, Firefox, Edge, Safari) that supports ES6 and web crypto APIs.

4. **Try a Different Browser:** To isolate if it's your browser's configuration, try the same page in another browser without added extensions.
5. **Look at Console Errors:** If you know how, open the browser Developer Tools (press F12 or Cmd+Option+I), check the Console tab for JS errors.
6. **Check Device Resources:** If your device is very slow or low on battery, it might not execute the Proof-of-Work efficiently; plug in or restart if needed.

Why not simply disable these protections?

It's important to mention that telling users to "bypass" protections isn't the best advice. The Proof-of-Work challenge helps keep websites safe and fast for everyone— including you. The fix is to enable or support the required JS and modern features, not to avoid the protection.

Summary Table: Why Anubis Shows Speed: 0kH/s

Cause	Explanation	Suggested Fix
JavaScript Disabled	Browser can't execute the computing script	Enable JavaScript in browser settings
Script-Blocking Extensions	Privacy or ad blockers stopping PoW script	Temporarily disable extensions and refresh page
Outdated Browser	Lacks modern JS/crypto APIs	Update your browser to the latest version
Network or Policy Restrictions	Firewall or proxy blocks JavaScript or crypto calls	Try a different network or consult IT about exceptions
Low Device Performance	Device too slow to compute PoW in reasonable time	Restart device or use a more powerful device if possible

Conclusion

Seeing **Speed: 0kH/s** stuck forever on Anubis anti-bot challenges is a sign your browser is blocked from properly running the JavaScript Proof-of-Work that verifies you're a real user. This system, inspired by Hashcash, asks your device to do a modest amount of computational effort to filter out malicious bots and maintain website reliability.

The fastest fix is to ensure your browser supports and can execute JavaScript with modern features, by enabling JavaScript, updating your browser, disabling interfering extensions, and checking your network settings.

Remember, these steps aren't about defeating protections—they help the welcome you come through the gates quickly, safely, and fairly.

By understanding what's happening and why Anubis reports "Speed: 0kH/s," you'll be better equipped to fix the issue and enjoy smooth, secure browsing.