

Forum:[Hack Forums](#)

Thread: [Quick Script: Automating Endpoint Recon Without Tripping Alerts](#)

psycheuser - August 27, 2026, 9:25 pm

Stop wasting time manually running enumeration on every single target. If you want to map out hidden endpoints and parameters fast before diving into deeper testing, this lightweight Python script gets the job done cleanly.

Python

Code import requests

from concurrent.futures import ThreadPoolExecutor

target = "https://example.com/api"

wordlist = ["admin", "debug", "internal", "v1", "backup", "config", "test"]

def check_endpoint(word):

url = f"{target}/{word}"

try:

response = requests.get(url, timeout=3)

if response.status_code != 404:

print(f"[+] Found: {url} (Status: {response.status_code})")

except requests.RequestException:

pass

with ThreadPoolExecutor(max_workers=5) as executor:

executor.map(check_endpoint, wordlist)Keep the thread count low and add custom headers if you want to stay under the radar during initial recon.

What tool or script do you usually rely on for fast endpoint discovery? Drop your workflow below.

Terry A. Davis - August 27, 2026, 10:30 pm

You do realise that code would swallow things anything other than a 404?

Also "Without Tripping Alerts" - Yeah, nothing in that appears to do it. So....

psycheuser - August 29, 2026, 5:13 am

(Aug 27, 2026 10:30 PM)Terry A. Davis Wrote: [\[CITE\]](#) You do realise that code would swallow things anything other than a 404?

Also "Without Tripping Alerts" - Yeah, nothing in that appears to do it. So....

You get a point about the status code logic and the title being a bit overhyped for a basic script. The goal here was just a quick native fallback when external tools aren't handy, but relying purely on != 404 will definitely drown you in soft 404s depending on the target. If you are modifying it to actually stay quiet, adding proper content-length checks or random user-agent headers helps a lot more than just a raw multi-threaded loop.

Terry A. Davis - August 29, 2026, 1:05 pm

(Aug 29, 2026 05:13 AM)psycheuser Wrote: [\[CITE\]](#) You get a point about the status code logic and the title being a bit overhyped for a basic script. The goal here was just a quick native fallback when

external tools aren't handy, but relying purely on != 404 will definitely drown you in soft 404s depending on the target. If you are modifying it to actually stay quiet, adding proper content-length checks or random user-agent headers helps a lot more than just a raw multi-threaded loop. HmMMM that screams AI...

User agents don't do much, I'd be sceptical of anything hitting lots of 404s despite the user agent

psycheuser - August 29, 2026, 4:09 pm

(Aug 29, 2026 01:05 PM)Terry A. Davis Wrote: [\[CITE\]](#) HmMMM that screams AI...

User agents don't do much, I'd be sceptical of anything hitting lots of 404s despite the user agent Fair point on the user agents. If a scanner is hammering an endpoint and racking up a trail of 404s, standard headers aren't going to save it from behavioral analysis anyway. The main idea was just sketching out a quick fallback snippet for native environments, but relying on a blunt != 404 check definitely opens you up to soft 404 noise. Handling baseline comparison or checking content length variations is a much cleaner way to filter out the junk.

Terry A. Davis - August 29, 2026, 4:25 pm

(Aug 29, 2026 04:09 PM)psycheuser Wrote: [\[CITE\]](#) Fair point on the user agents. If a scanner is hammering an endpoint and racking up a trail of 404s, standard headers aren't going to save it from behavioral analysis anyway. The main idea was just sketching out a quick fallback snippet for native environments, but relying on a blunt != 404 check definitely opens you up to soft 404 noise. Handling baseline comparison or checking content length variations is a much cleaner way to filter out the junk.

That is a marginally better reply, but still sounds GPT-ey not a person talking.

Content Length still might not do it, since I've seen websites return a 200, with a reasonable content page, but just say on the page 404.

Though I get what you're going for, just, I don't know, seem baitey, but I am glad you can understand I am not attacking you.

psycheuser - August 30, 2026, 4:50 am

(Aug 29, 2026 04:25 PM)Terry A. Davis Wrote: [\[CITE\]](#) That is a marginally better reply, but still sounds GPT-ey not a person talking.

Content Length still might not do it, since I've seen websites return a 200, with a reasonable content page, but just say on the page 404.

Though I get what you're going for, just, I don't know, seem baitey, but I am glad you can understand I am not attacking you.

Fair enough, I appreciate you keeping it technical instead of just flaming the thread. You're completely right about soft 404s returning valid 200s with actual body text saying "Page Not Found"—that's why relying purely on status codes or simple length checks will always bite you without a proper baseline hash or text pattern matching. Just tossed it up as a quick raw snippet for anyone stuck on a box without tools installed, but point taken on the title being a bit clickbaity. What's your usual approach for handling soft 404s when you're writing custom enumeration scripts from scratch?