GAJWAF · OPERATIONS PLAYBOOK

Tuning the bot challenge

Stop bots — without trapping real users in a loop

Bind to the device. Exempt what can't solve it.

Get the browser check to stop bots **without** trapping real browsers, APIs or privacy-relay users in a loop. This playbook covers the settings that matter and the four situations that come up most.

7 settings
On the Bot check tab

4 scenarios
With cause and fix

1st check
Always the cluster secret

GajShield Infotech Private Limited
Mumbai, India

Version 1.0 · October 2026
For GAJWAF administrators & analysts

LEGAL NOTICE

© 2026 GajShield Infotech Private Limited All rights reserved. This document is provided for the use of GAJWAF customers and partners. No part of it may be reproduced, distributed or transmitted in any form or by any means without the prior written permission of GajShield Infotech Private Limited

The information in this document is provided "as is", without warranty of any kind, express or implied, and is subject to change without notice. Features, menu labels and default values may differ between GAJWAF releases. While every effort has been made to ensure its accuracy, GajShield Infotech Private Limited accepts no liability for any loss, damage or service disruption arising from the use of, or reliance on, this document. This playbook is general operational guidance. It does not replace your organisation's own incident-response or change-control procedures, and GAJWAF is not a substitute for network-layer (L3/L4) DDoS protection.

GAJWAF and GajShield are trademarks of GajShield Infotech Private Limited Amazon Web Services, AWS and AWS Shield are trademarks of Amazon.com, Inc. or its affiliates. Microsoft and Azure are trademarks of the Microsoft group of companies. All other product and company names mentioned in this document are trademarks of their respective owners; they are used for identification purposes only, and their mention does not imply endorsement.

AT A GLANCE

Three rules for a quiet browser check

GOLDEN RULE

The browser check only runs in **Block** mode. **Bind passes to the device**, **exempt the clients that can't solve a check**, and if users are looping, **suspect a multi-node secret mismatch** before you touch anything else.

1
Block mode only
The check never runs in Detect. If nobody is being challenged, check the mode first.

2
Bind to the device
A pass tied to the browser survives rotating IPs, proxies and privacy relays.

3
Exempt what can't solve it
APIs, webhooks, health checks and mobile apps go on the exempt-path list.

IMPORTANT

Users looping? Check, **in this order**: (1) cluster secret shared on every node → (2) device binding on → (3) "Pass valid for" long enough.

✕ DON'T

- **Don't** debug a re-challenge loop before ruling out a cluster secret mismatch.
- **Don't** IP-ban privacy-relay or corporate-proxy ranges — bind to the device.
- **Don't** leave debug logging on permanently — it's verbose.

WHO **Admin** (config) + **analyst** (under-attack, debug logging).

CONTENTS

1 The settings that matter 3
2 Users challenged over and over 3
3 API, mobile or server-to-server clients get challenged 4
4 Users behind a corporate proxy or privacy relay 4
5 "bot-biometrics" Medium events everywhere 5
✓ Verify & tuning record 5

© 2026 GajShield Infotech Private Limited All rights reserved.

REFERENCE

1 The settings that matter

REFERENCE WHO Admin

WHERE **Edit configuration → All settings → Bot check**

Console label	Config key	What it does — and the usual setting
"Check browsers"	`challenge.mode`	`off`, `suspicious` or `always`. The check **only challenges in Block mode** — Detect never challenges.
"Identify devices with a WAF cookie"	`tracking.device_id`	Binds a pass to the **browser** (a device cookie), not the IP. **Essential** for users on rotating IPs — keep it **on**.
"Pass valid for"	`challenge.ttl`	How long a solved check lasts. Default **12 h**.
"Never check these paths"	`challenge.exempt_paths`	Path prefixes for **APIs, webhooks, health checks, mobile endpoints**.
"Clients that cannot run the check"	`challenge.non_browser`	**allow** — keep APIs and apps working; **block** — lock down while everyone is checked.
Under-attack mode	site tile → **"Under attack…"**	Challenge **every** visitor for a set duration (see *Site under active attack*).
"Debug logging"	`challenge.debug`	Verbose diagnostics for **GajShield Support**. Turn on only when needed — and off afterwards.

> NOTE
> Clients with **valid credentials** (API authentication or the sign-in gateway) are identified and **never** challenged.

SCENARIO 1 OF 4

2 Users challenged over and over

SCENARIO WHO Admin

YOU SEE A user passes the browser check, but is asked again — it **never sticks**.

Check	Likely cause	Fix
1st	**Several nodes don't share the same cluster secret.** By far the most common cause.	Fix the cluster first — *Multi-node cluster setup & verification* playbook.
2nd	**Device binding is off** — the pass is tied to an IP that keeps changing.	Turn on **"Identify devices with a WAF cookie"**.
3rd	**The pass expires too soon.**	Lengthen **"Pass valid for"**.

© 2026 GajShield Infotech Private Limited All rights reserved.

> **IMPORTANT**
> Don't debug a re-challenge loop before ruling out a **cluster secret mismatch**.

> **NOTE**
> Still looping after all three? Turn on **"Debug logging"**, reproduce the problem, note the **time** and the **user's IP**, send them to **GajShield Support** — then turn debug logging off. The log records, for every navigation, why it was or wasn't challenged.

SCENARIO 2 OF 4

3 API, mobile or server-to-server clients get challenged

SCENARIO WHO Admin

YOU SEE Integrations, mobile apps or webhooks start failing once the browser check is on.

Check	Likely cause	Fix
1	Their paths aren't exempt.	Add the path prefixes to **"Never check these paths"**.
2	They don't present credentials.	Have them send **valid credentials** — authenticated clients are never challenged.
3	They **can't run JavaScript**.	Set **"Clients that cannot run the check"** to **allow**.

SCENARIO 3 OF 4

4 Users behind a corporate proxy or privacy relay

SCENARIO WHO Admin + analyst

Rotating egress IPs look like a new client on every request — for example Apple **iCloud Private Relay** (`172.224.x` / `172.225.x`) or a large corporate proxy pool.

Do	Don't
Turn on **device binding** — "Identify devices with a WAF cookie". The pass then **follows the browser**, whatever IP it arrives from.	**IP-ban** these ranges — you'd block every real user behind them.

SCENARIO 4 OF 4

5 "bot-biometrics" Medium events everywhere

SCENARIO WHO Analyst

Benign. A human navigated without moving the mouse. It is **informational** and **never** bans on hosting or relay networks.

© 2026 GajShield Infotech Private Limited All rights reserved.

> **TIP**
> No action needed. Only look further if you **also** see the re-challenge loop in Scenario 1.

BEFORE YOU CLOSE THE CHANGE

✓ Verify & tuning record

☐	**Real browsers**	A real browser solves the check **once** and isn't asked again — **including across nodes** behind a load balancer.
☐	**APIs & apps**	API, mobile and server-to-server clients work without being challenged.
☐	**Bots stopped**	Bots and scripted clients are stopped.
☐	**Debug off**	"Debug logging" is turned off again.

TUNING RECORD

Site	
Problem reported	☐ Re-challenge loop ☐ APIs challenged ☐ Proxy / relay ☐ Biometrics
Cluster secret checked	☐ Single node ☐ Shared on all nodes
Settings changed (from → to)	
Paths exempted	
Debug logging used / turned off	☐ Not used ☐ Used and turned off
Done by / date	

Related documents

Document	Use it for
Multi-node cluster setup & verification (playbook)	Ruling out the #1 cause of re-challenge loops.
Site under active attack · DDoS / Layer-7 flood (playbooks)	Using under-attack mode against attacks and floods.
GAJWAF Troubleshooting & Support Manual	Chapter D — bot check and challenge problems.
GAJWAF Features & Configuration Guide	The Bot check tab (chapter 7).

© 2026 GajShield Infotech Private Limited All rights reserved.