

GAJSHIELD · WEB APPLICATION & API FIREWALL

GAJWAF

Quick Setup Guide

Deploying from the AWS & Azure Marketplace

Your first 15 minutes, from launch to blocking attacks

This guide takes you from a freshly launched GAJWAF instance to a protected, HTTPS-enabled site in blocking mode — securing the console, proxying one site in watch-only mode, installing a certificate, and switching on protection.

~15 min

Time to first protected site

10 steps

From sign-in to updates

AWS · Azure

Marketplace images

About this document

Document	GAJWAF Quick Setup Guide — AWS & Azure
Version	1.0 · October 2026
Applies to	GAJWAF — AWS Marketplace, Azure Marketplace and BYOL editions
Published by	GajShield Infotech Private Limited, Mumbai, India

COPYRIGHT

© 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

DISCLAIMER

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.

USE OF THIS INFORMATION

Apply configuration changes in accordance with your organisation's change-control procedures, and validate them in your own environment before relying on them in production.

TRADEMARKS

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.

IN THIS GUIDE

Contents

0	What you deployed	5
1	Sign in for the first time	5
2	Confirm management access is locked down	6
3	Add your first site	6
4	Send traffic through the WAF and test it	8
5	Install your HTTPS certificate	8
6	Watch, then switch to blocking	9
7	Turn on the protections you want	10
8	Secure the console properly	10
9	Keep it updated	11
★	Quick reference	11

CHECKLIST

Before you begin

- A GAJWAF instance launched from the **AWS** or **Azure Marketplace**, with its public IP.
- Access to your cloud console to read the **Instance ID** (AWS) or **vmId** (Azure).
- Your management IP / CIDR (office or VPN) for console access.
- The host names of the site to protect, and the address of its backend web server.
- Optionally, a certificate (full chain) and private key in PEM format.

AT A GLANCE

Highlights

GAJWAF is a reverse-proxy web application & API firewall (WAAP): traffic reaches GAJWAF first, it inspects every request, and forwards the clean ones to your web server.

Reverse-proxy WAAP

Traffic reaches GAJWAF first; every request is inspected and only clean traffic is forwarded to your web server.

Two inputs per site

Each site needs just its **host names** and the **backend** server to send good traffic to.

Risk-free start

Watch only (detect) mode shows what would be blocked without touching real users.

One-click security levels

A single choice sets blocking strength, bot checks, login protection and data-leak handling.

HTTPS your way

Upload your own certificate, or let GAJWAF obtain and renew one from Let's Encrypt.

In-place updates

Signed, encrypted packages applied from the console — no new VM image or AMI required.

THE SETUP PATH

Four stages from launch to protection

STAGE 1

Secure

Sign in, lock down port 9443

STAGE 2

Proxy

Add a site in detect mode, route DNS

STAGE 3

Encrypt

Install your HTTPS certificate

STAGE 4

Enforce

Review events, switch to blocking

IMPORTANT

Never expose the management console (port 9443) to the Internet, and make sure your backend accepts traffic only from GAJWAF so the WAF cannot be bypassed.

0 What you deployed

GAJWAF sits in front of your web servers. Clients connect to GAJWAF on the standard web ports, and you manage it on a separate console port:

Port	Purpose	Who should reach it
80 / 443	Protected web traffic (your clients)	The Internet
9443	The management console	Only you (office / VPN IP)

NOTE

The Marketplace templates already restrict port 9443 to the management network you supplied. If you launched the instance by hand, set this yourself (see Step 2).

1 Sign in for the first time

Open `https://<public-ip>:9443/`. The console uses a fresh self-signed certificate, so your browser warns once — proceed.

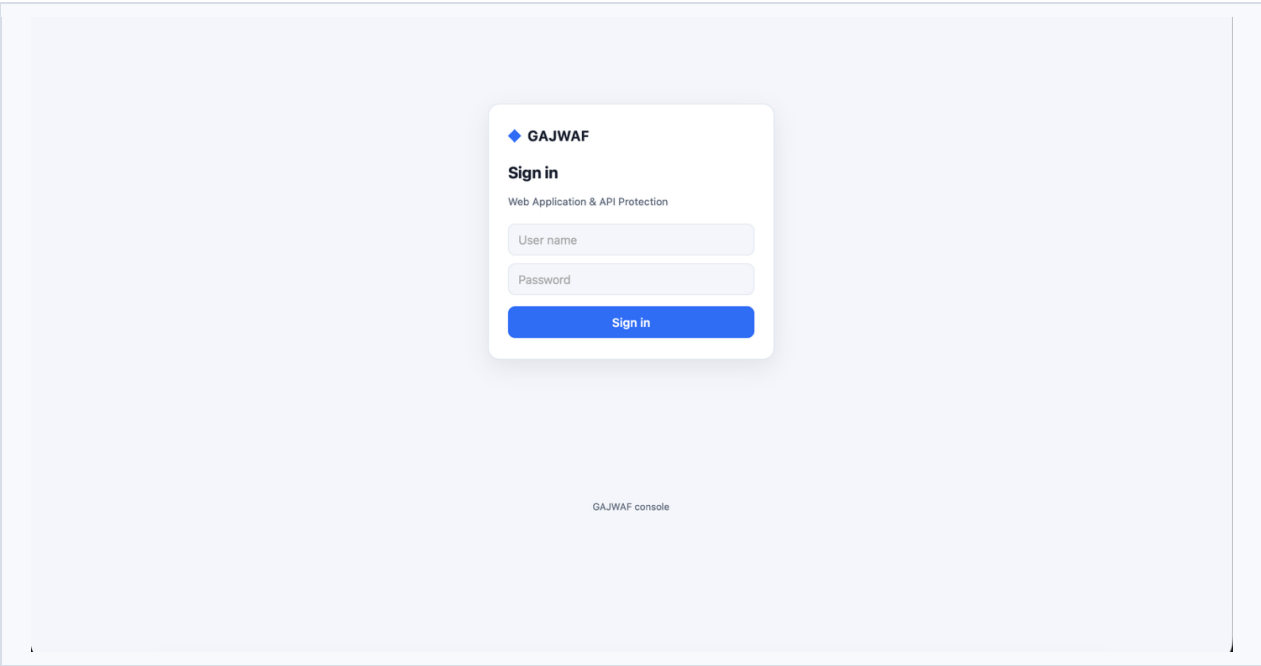


Figure 1 GAJWAF console sign-in page

Sign in with:

Field	Value
User	admin
Password (AWS)	The EC2 Instance ID, e.g. i-0abc123def4567890
Password (Azure)	The VM's vmId (a GUID, under the VM's Properties)

Both values can be read from your cloud console — no SSH needed. You are forced to set a new password immediately; choose a strong one.

2 Confirm management access is locked down

Check the firewall in front of the instance — the **Security Group** on AWS, or the **Network Security Group (NSG)** on Azure. It should allow:

- **80/443** from anywhere
- **9443** only from your management IP/CIDR

IMPORTANT

If the console port is open to the Internet, tighten it now. Port 9443 should never face the Internet.

Also make sure the application can only be reached **through** GAJWAF: add a rule on the backend's subnet that allows inbound traffic only from the WAF's private IP, so nobody can bypass the WAF and hit the app directly.

3 Add your first site

Go to **Sites** → **Add site**. The simple form asks for only what matters:

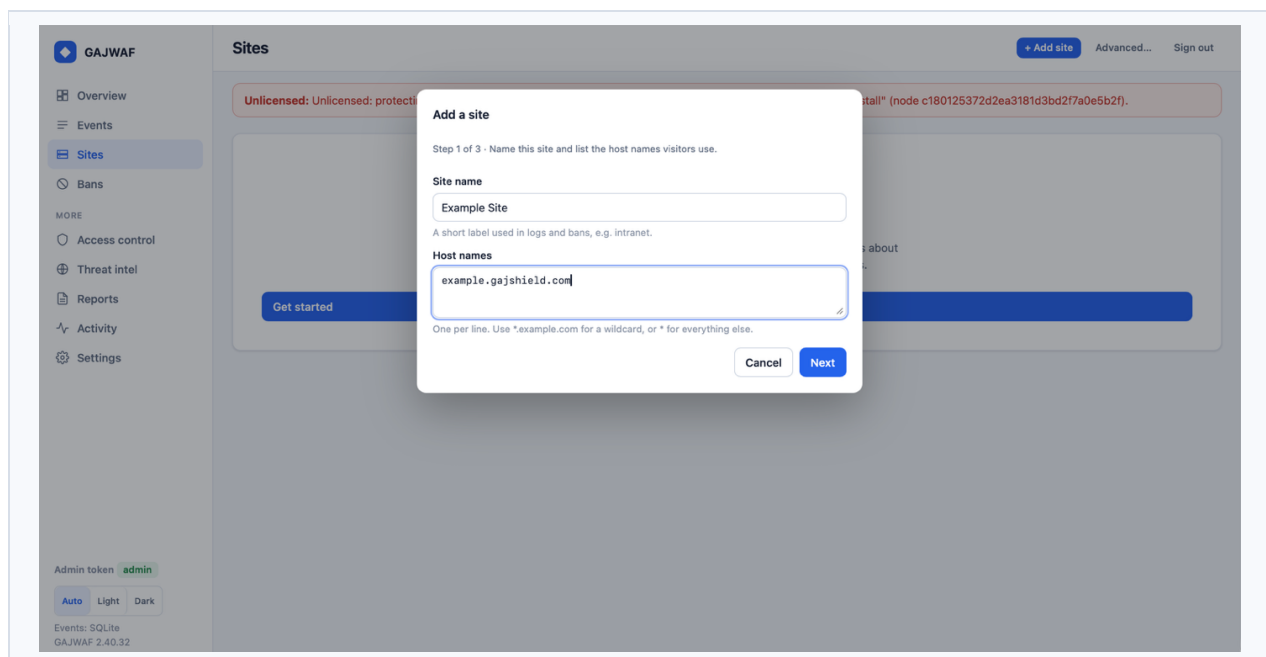


Figure 2 Sites → Add site

1. **Site name** — a short label used in logs, e.g. `web`.
2. **Host names** — the domains this site answers for, one per line (`www.example.com`, `example.com`). Use `*.example.com` for a wildcard, or `*` to catch everything else.

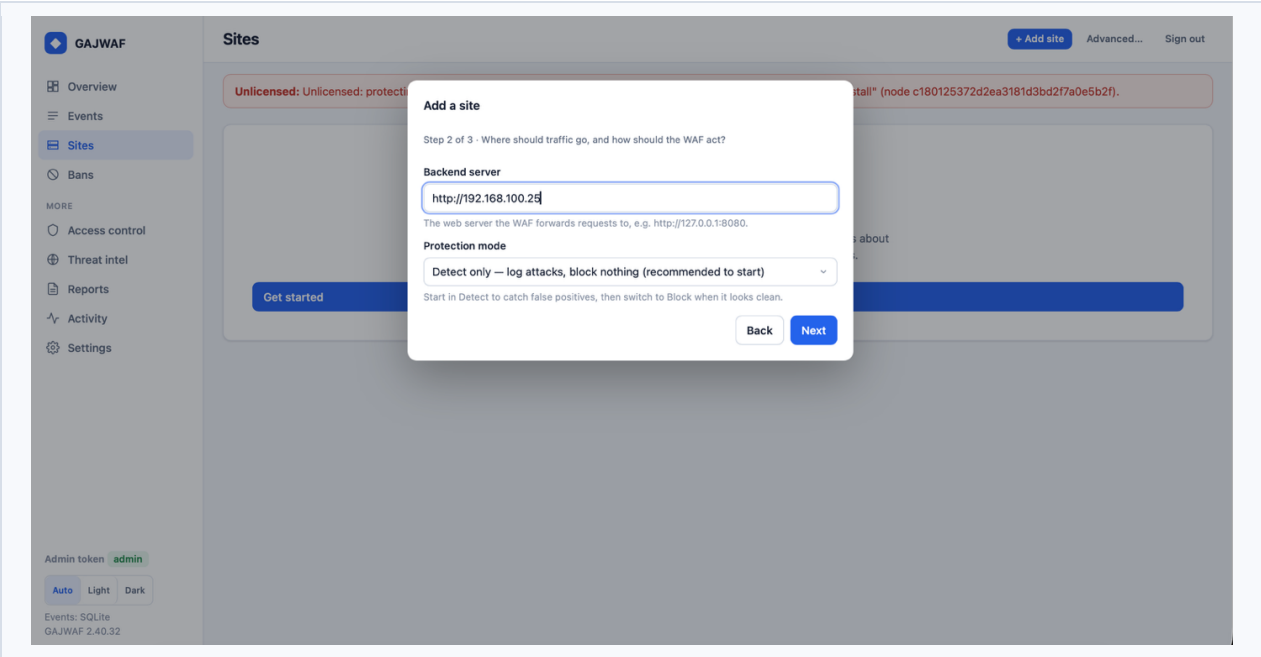


Figure 3 Entering host names for the site

- 3. **Your web server (backend)** — where clean traffic goes, e.g. `http://192.168.100.25`. Add more lines to load-balance across servers.
- 4. **Protection mode** — start on **Watch only (detect)**. This proves traffic flows and shows what *would* be blocked, with zero risk to real users.

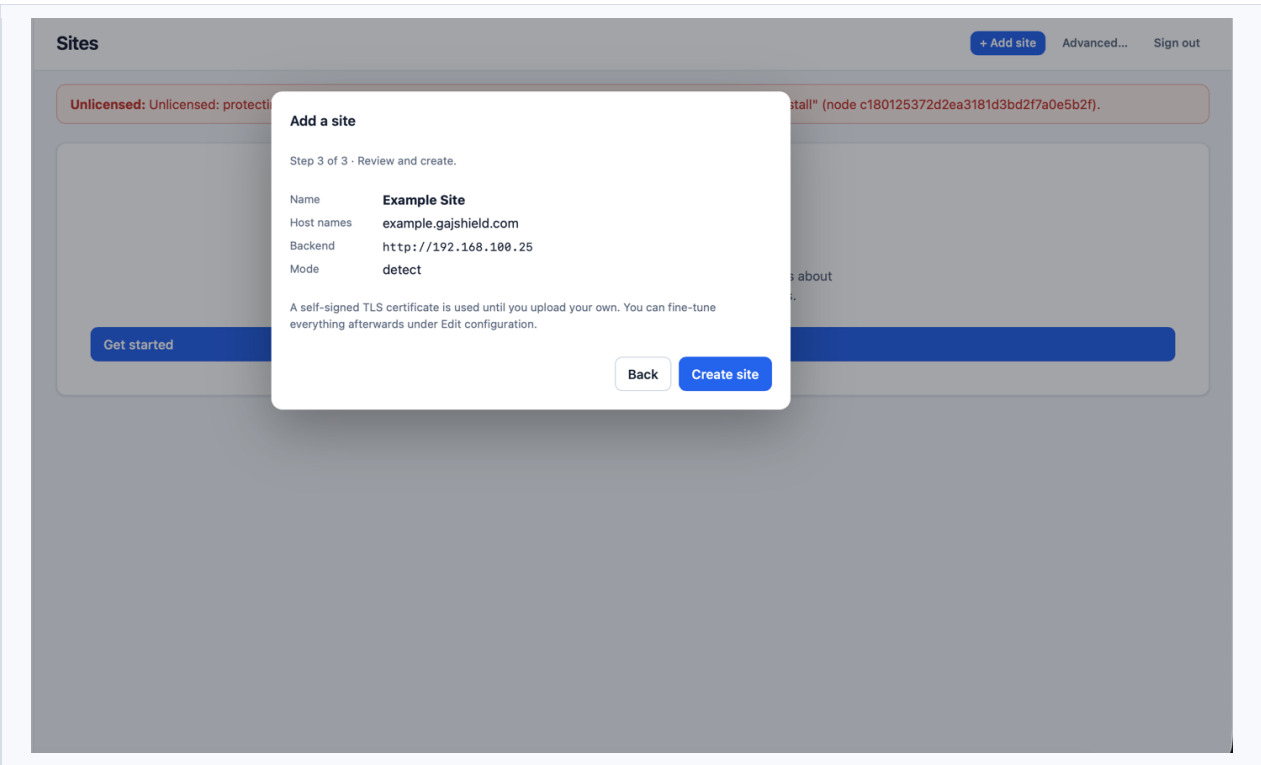


Figure 4 Review and create the site

Figure 5 Site editor — security level, host names, backend and protection mode

Pick a **security level** from the cards at the top — it sets blocking strength, bot checks, login protection and data-leak handling in one step — then click **Save**.

NOTE

Requests for host names or IPs you have not configured get a **421 Misdirected Request**. That is GAJWAF correctly refusing traffic it does not recognise — mostly Internet scanners hitting the raw IP.

4 Send traffic through the WAF and test it

Point your application's DNS at the WAF's public IP — or place the WAF behind your existing load balancer / Azure Front Door and forward 80/443 to it.

In the site editor, use **Test connection** to confirm GAJWAF can reach your backend. Then load your site in a browser: it should serve normally, and the requests should appear under **Overview** and **Events**.

5 Install your HTTPS certificate

New sites use a self-signed certificate until you add a real one.

1. Edit the site and open **HTTPS certificate**.
2. Paste or upload your **certificate (full chain)** and **private key (PEM)**.
3. Click **Validate & install**.

The certificate is checked before anything changes, and installing it leaves the rest of your settings untouched.

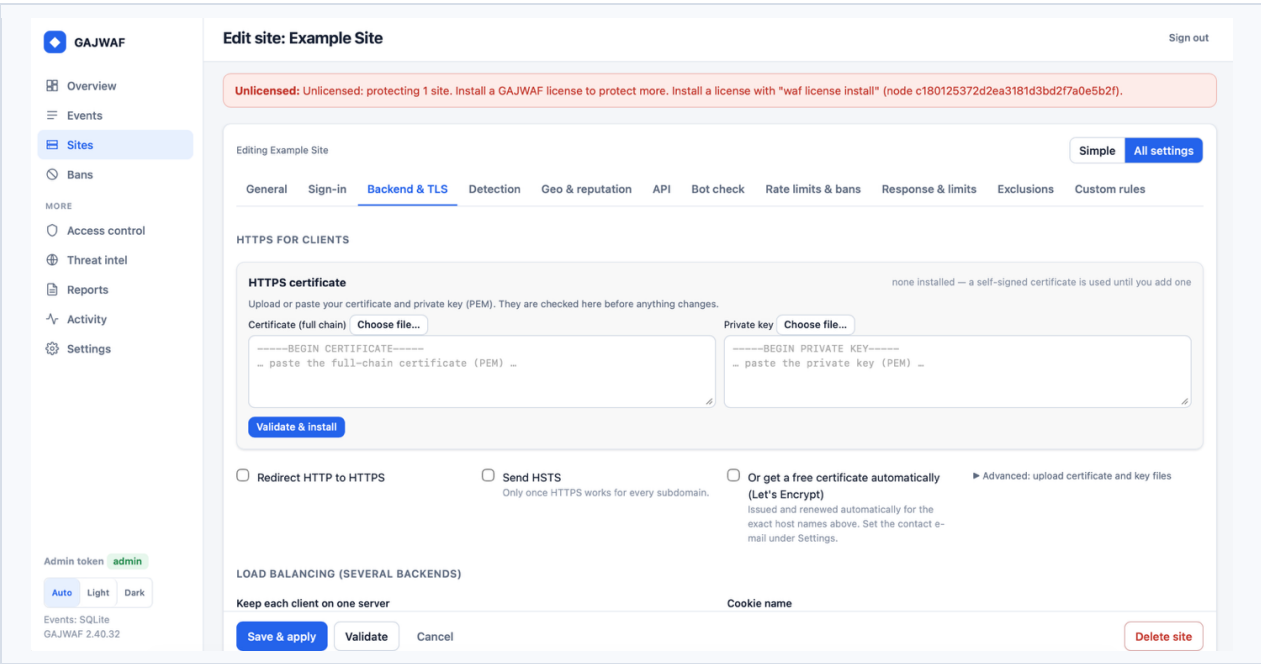


Figure 6 Installing an HTTPS certificate

TIP

For an Internet-facing site on ports 80/443, GAJWAF can obtain and renew a **Let's Encrypt** certificate automatically. Enable it under *All settings* → *Backend & TLS* (tls.acme).

Use either a Let's Encrypt certificate or an uploaded one — not both.

6 Watch, then switch to blocking

Leave the site in **detect** mode for a while — an hour, a day, your call — and review **Events** and the **Overview** dashboard:

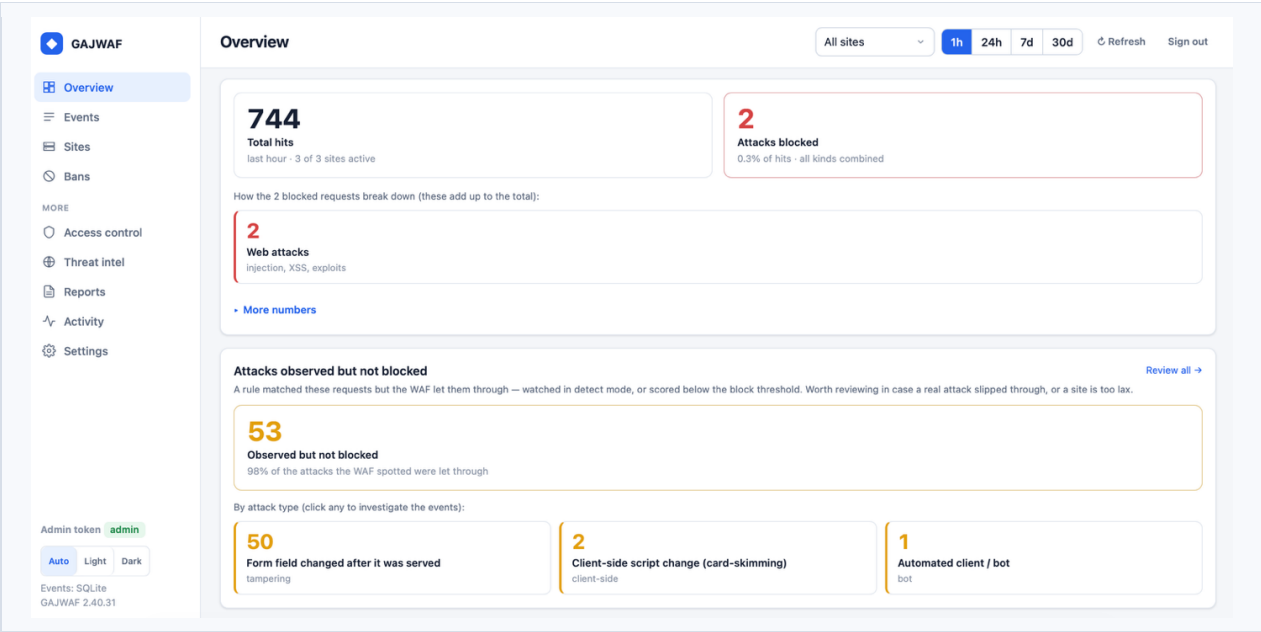


Figure 7 Overview dashboard

- The **Attacks observed but not blocked** panel shows exactly what GAJWAF *would* have stopped. If those are all genuine attacks and scanners, you are safe to block.
- If a legitimate request was flagged, open the event and use **Not an attack?** to add a precise exclusion (or lower the site's security level).

TIP

When it looks clean, edit the site and set **Protection mode** → **Block attacks**. From then on, attacks are stopped and everything else passes straight through.

7 Turn on the protections you want

These live in the site editor under **All settings** (the toggle at the top). The security level you picked has already enabled the common ones.

Protection	What it does
Data-leak prevention	Under <i>Response & limits</i> : masks payment cards, Aadhaar/PAN and US SSNs, and redacts stack traces, database errors and private keys if your app ever leaks them in a response.
Bot check	A lightweight browser check that stops scripted floods and headless bots. Start with <i>Suspicious clients only</i> , or <i>Every visitor</i> when under attack.
Rate limits & bans	Caps requests per client and per login endpoint.
API	Upload an OpenAPI spec to enforce your API contract and block off-schema requests.
Sign-in (zero-trust access)	Put a login in front of an internal app, or read your app's existing identity — use the <i>Set up zero-trust access</i> wizard on the Sign-in tab.

8 Secure the console properly

Before you call it done:

1. **Create real console users** under **Settings** → **Console users** (one per operator), and stop relying on the seeded admin account and the bootstrap token.
2. Enable **require login** and **two-factor authentication** for the console.
3. If you need more than one protected site, install your GAJWAF license under **Settings** → **License** (or **waf license install**).

NOTE

On a paid **AWS Marketplace** listing, the instance is billed per hour by AWS with no site cap. On **BYOL / Azure**, the per-site license governs how many sites you may run.

9 Keep it updated

Apply GAIWAF updates in the console under **Settings** → **Software update** (signed, encrypted packages). No new VM image or AMI is needed for a version bump.

★ Quick reference

Item	Where / value
Console URL	<code>https://<public-ip>:9443/</code>
First sign-in	User <code>admin</code> ; password = AWS Instance ID or Azure vmId
Add a site	Sites → Add site
Test backend reachability	Site editor → Test connection
Install certificate	Edit site → HTTPS certificate → Validate & install
Automatic Let's Encrypt	<i>All settings</i> → <i>Backend & TLS</i> (<code>tls.acme</code>)
Go live	Edit site → Protection mode → Block attacks
Console users / 2FA	Settings → Console users
License	Settings → License or <code>waf license install</code>
Updates	Settings → Software update