



AIRPORT CYBER RESILIENCE SIMULATOR

VERSION 2 · SETUP & CONFIGURATION GUIDE

A step-by-step guide to deploying, configuring and operating the containerised kiosk simulator —
from Docker Compose to your first live attack scenario.



OVERVIEW

What you are deploying

The Airport Cyber Resilience Simulator is a self-contained, kiosk-grade training and exhibition tool. A Python (FastAPI) engine runs a per-second simulation loop and streams live state over WebSockets to an immersive React command-center dashboard. Everything is data-driven and fully containerised — one command brings the whole stack up.

Layer	Technology
Backend	Python 3.11 · FastAPI · asyncio · WebSockets
Frontend	React + GSAP + Web Audio (all vendored, no CDN needed)
Delivery	Docker + Docker Compose (single service, port 8000)
Data	29 UK airports · 26 nodes · 9 attack playbooks (JSON)
Compliance	NCSC CAF objectives + UK Cyber Security and Resilience Bill

DEPLOYMENT

Setup

1 Confirm prerequisites

Install Docker Desktop (Windows/macOS) or Docker Engine + Compose (Linux). This is the only requirement — Python and all frontend libraries ship inside the image.

Verify the toolchain is present and the daemon is running:

```
docker --version
docker compose version
docker info
```

No Docker? You can also run it directly with Python 3.11+: `pip install -r backend/requirements.txt` then `uvicorn main:app --port 8000`.

2 Get the project files

Place the `airport-cyber-sim-v2/` folder anywhere on the host. The structure you should see:

```
airport-cyber-sim-v2/
docker-compose.yml sim.ps1 / Makefile README.md
backend/
Dockerfile main.py requirements.txt
data/ airports.json infrastructure.json playbooks.json
static/ index.html app.js styles.css vendor/
```

3 Launch with Docker Compose

From the project root, build and start the container:

```
cd airport-cyber-sim-v2
docker compose up --build
```

The first build downloads the Python base image and installs dependencies (about a minute). When you see **Uvicorn running on 0.0.0.0:8000**, the simulator is live.

Prefer shorthand? Use the bundled helpers:

```
.\sim.ps1 up # Windows PowerShell
make up # macOS / Linux / WSL
```

4 Open the kiosk

Browse to the address below on the kiosk display. For a true full-screen exhibition kiosk, press F11 (or launch Chrome/Edge with `--kiosk http://localhost:8000`).

```
http://localhost:8000
```

The stage is locked to a 1920×1080 layout and auto-scales to fit any screen, so it looks correct on portrait tablets and 4K panels alike.

OPERATING THE KIOSK

Configuration & first run

5 Profile the airport

On the Initial Configuration Screen, choose one of the 29 UK passenger airports. Your selection scales the entire economic model — Heathrow drives the highest revenue-per-second and maximum statutory fines; regional airports scale down proportionally.

Toggle Enable Simulation Audio to match your stand's noise environment, then press INITIALIZE SIMULATOR to enter the dashboard.

6 Arm and launch a playbook

In the left sidebar, pick one of the 9 attack playbooks, then press START SIMULATION. A dramatic 5-second countdown plays; the simulation clock, financial bleed and the point-of-ingress indicator only begin when it hits zero.

7 Respond to the incident

Tap any node to open its overlay — service detail, live capacity, threat vector, mitigation options and a Regulatory Insights tab mapping the node to its CAF objective and statutory reporting duty. Choose a mitigation deliberately; each carries an operational trade-off:

Mitigation	Effect on spread	Capacity	Cost
1 · Isolate Node	Severs links, halts spread	→ 0% (runway lights go dark)	Max operational bleed
2 · Reduce / Throttle	Slows spread by 80%	→ 40% (lights dim & flicker)	Moderate
3 · Stop Attack (Patch)	5s patch race — reverts to normal only if it beats the attack to the next node	Unchanged	£50k flat fee

8 Work the Crisis Communications module

Once an attack begins you have a 15-second window to Release a Proactive Media Statement, halving the reputational-loss multiplier. Miss it and the phone and email icons flash red — answer them to hold the baseline penalty. Ignore the barrage and a communications blackout triples the reputational bleed.

9 Reset between visitors

Press RESET TO CONFIG to clear all state and return to airport selection. If the kiosk is idle for 90 seconds it auto-resets to an animated attract loop, ready for the next visitor.

UNDERSTANDING THE METRICS

The math engine

Every second the backend recomputes total loss and streams it to the odometer, which shifts colour white → amber → neon-red as it climbs:

```
Total_Loss = Operational_Cost + Regulatory_Fines + Mitigation_Costs

Operational_Cost += Revenue_Per_Sec x Node_Impact x Downtime_Penalty
x Reputational_Multiplier (every second)
Regulatory_Fines += Node_Flat_Fine x Airport_Fine_Multiplier
(on breach of a data-sensitive node)
Mitigation_Costs += 50,000 (per emergency patch deployed)
```

- Downtime penalties: Normal 0.0 · Throttled 0.6 · Isolated 1.0.
- Impact factors are per-node (ATC ≈ 0.85, Power ≈ 0.88, Retail POS ≈ 0.10).
- Reputational multiplier: 0.5 proactive · 1.0 baseline · 3.0 blackout.

REFERENCE

Customise & troubleshoot

The simulator is fully data-driven — extend it by editing the JSON in backend/data/; the compose file mounts that folder read-only so changes hot-reload without a rebuild.

Task	Where
Add an airport / change scalars	backend/data/airports.json
Add nodes, links or impact factors	backend/data/infrastructure.json
Add or edit attack playbooks	backend/data/playbooks.json
Port 8000 already in use	Edit the ports mapping in docker-compose.yml
Check container health / logs	sim.ps1 status · sim.ps1 logs
Stop everything	docker compose down · sim.ps1 down