NetSimIT

Facilitator Guide

Running a session, end to end

Gamified Network Attack Simulation Platform

v1.0

- The cost of a single gap: the auto-cascade mechanic viscerally demonstrates how quickly one missing control turns into a full breach.
- Judgement under uncertainty: the asset-response actions are deliberately probabilistic and unlabelled with exact numbers in the player-facing UI — participants have to make a call without a guaranteed outcome, just like in reality.
- Crisis communications timing: internal, external, and regulatory disclosure decisions are graded on timing and sequencing, not just on whether they were sent at all.
- Compliance is not abstract: every outcome ties back to a named, real-world-style framework (NCSC CAF, Cyber Essentials Plus, NIST CSF 2.0, SOC 2, and a stand-in Cyber Resilience Bill).

1. Deployment

NetSimIT is a single self-contained HTML file with no backend and no data persistence — nothing a participant does is saved anywhere, including between browser refreshes. It can be run in two ways.

1.1 Directly in a browser

Open `index.html` directly in any modern browser (Chrome, Edge, or Firefox). No server, install, or internet connection is required beyond the one-time load of the Tailwind CSS CDN script referenced in the file's head.

1.2 Docker (recommended for a shared or classroom session)

A Dockerfile and docker-compose.yml are included alongside the application file, serving it via nginx:alpine on port 8080.

```
FROM nginx:alpine

COPY index.html /usr/share/nginx/html/index.html

EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
```

```
services:
  netsimit:
    build: .
    container_name: netsimit
    ports:
      - "8080:80"
    restart: unless-stopped
```

From the project directory, run `docker compose up --build` and open `http://localhost:8080`. Any device on the same network can reach it at the host machine's address on port 8080, which is the simplest way to put it on a shared screen or let a room of participants each run their own instance from laptops.

2. Suggested Session Structure

A single pass through one playbook takes two to five minutes depending on how much the group discusses each choice. A typical 60–75 minute session might run as follows:

Time	Activity
5–10 min	Introduce the tool, the topology diagram, and the six defensive posture controls.
10 min	Run one playbook together with every control switched off — let the group watch a full cascade to a breach.
10 min	Re-run the same playbook with the relevant controls switched on — compare outcomes directly.
15–20 min	Split into pairs or small groups, each running a different playbook and topology tier, using asset-response actions and crisis comms freely.
10–15 min	Regroup — each group shares their Session Debrief and talks through one decision they'd make differently.
10 min	Facilitator-led debrief: tie the in-app compliance framework references back to real obligations relevant to your organisation.

For a shorter slot (20–30 minutes), skip the small-group split and run two or three playbooks together as a full group, alternating who makes each choice.

3. How the Simulation Actually Works

Everything in NetSimIT is deterministic except the asset-response actions, which are probabilistic. Knowing the exact mechanics lets you answer "why did that happen?" accurately during debrief — this section is intentionally more detailed than the Player Guide, which keeps these numbers hidden to preserve the judgement-call framing.

3.1 Phase resolution

Each playbook has three phases; each phase has two choices; each choice is tied to exactly one of the five playbook-relevant controls (SOC, MDR, CTI, Vulnerability Management, Compliance Guard). When a choice is made, the simulator checks whether that specific control is switched on *at that exact moment* — not whether it was ever on, or will be on later. Toggling a control on mid-cascade genuinely changes the outcome of whichever phase resolves next.

3.2 The auto-cascade

If a phase is breached and it isn't the playbook's final phase, the simulator automatically resolves the next phase using its first available choice, evaluated against live toggle state, roughly every 1.5–2 seconds. This is the mechanism that makes "how quickly can this spread?" visible rather than theoretical. It only stops when: the next phase's control happens to be on, Incident Response is on, or a node action intervenes (below).

3.3 Incident Response (IR)

IR is the sixth toggle and behaves differently from the other five: it is never the reason a phase itself succeeds or fails, but if a phase is breached and IR is on, IR silently intercepts the cascade before it reaches the next phase. It has no effect on the final phase of a playbook (there is nothing left to intercept once the last phase itself has already failed) — a full breach on the final phase is always achievable regardless of IR.

3.4 Asset-response actions (probabilistic)

Clicking a node and choosing a response records that action against the node for the rest of the current playbook run. Whenever that node is actually targeted by a phase, and that phase's own control fails, the simulator rolls against the action's success chance as an *additional*, independent opportunity to contain it. Applying an action to an already-compromised node triggers an immediate roll instead, representing a reactive clean-up attempt.

Action	Success chance	Blocks further spread regardless of roll
Emergency Patch	75%	No
Isolate / Quarantine	85%	Yes — always
Reset Credentials & Sessions	45%	No
Enhanced Monitoring	20%	No
Restore from Clean Backup	70%	No
Take No Action	0%	No

Isolate / Quarantine is deliberately the standout option: high success chance *and* the only one that deterministically prevents the node being used as a pivot point even on a failed roll. This is intentional — it mirrors how disconnecting an asset is a blunt but highly reliable containment move in real incident response, at the cost of availability. Participants who over-rely on it will contain almost everything but should be prompted to consider the operational cost of constant isolation.

3.5 Crisis communications

Each of the three comms channels is judged once per playbook run against the incident's live stage at the moment it is pressed (no active incident / attack in progress / contained / contained by IR / full breach), plus, for external comms, whether internal comms were already sent, and for regulatory notification, whether a real compromise has actually occurred at all. A regulatory notification sent while the incident is still unfolding or immediately after containment passes the in-app Cyber Resilience Bill check; one sent only after a full breach fails it.

3.6 The Session Debrief

On Reset, if anything happened that session (any playbook run or comms decision), a randomised debrief is generated from session-wide history — not just the last playbook run — covering topology choice, final defensive posture, contained-versus-breached ratio, IR and asset-action usage, and comms timing. Multiple phrasings exist per category so it won't read identically across repeated sessions. It is generated fresh each time and is never saved — closing it discards it permanently, so encourage participants to read it before resetting.

4. Playbook Reference

All ten playbooks are always available; there is no difficulty gating. Use the table below to pick playbooks that match the audience or the point you want to make.

#	Playbook	Attack path	Mitigated by
1	Ransomware via Phishing	HR mailbox → Active Directory → Core Database	SOC / MDR
2	Supply Chain Compromise	DMZ Web Server → Core Database → Cloud Storage	Vulnerability Management
3	Insider Threat Data Theft	Core Database → HR Workstations → Cloud Storage	SOC / CTI
4	Cloud Misconfiguration Exploitation	Cloud Storage → Active Directory → Cloud Takeover	Vulnerability Mgmt / Compliance
5	Zero-Day Exploit on Edge Firewall	Edge Gateway → DMZ → Internal Reconnaissance	CTI / MDR
6	Business Email Compromise	HR/Finance → Active Directory → Wire Transfer	SOC / Compliance
7	DDoS Attack with Extortion	Edge Gateway → DMZ → Extortion Demand	MDR
8	State-Sponsored APT	DMZ → Active Directory → Edge Gateway (firmware)	CTI / SOC
9	IoT / OT Environment Breach	HR/Finance Subnet → Active Directory → Core Database	Vulnerability Mgmt / MDR
10	API Hijacking (BOLA)	DMZ → Core Database → Cloud Storage	Compliance / SOC

Facilitation notes

- **Ransomware via Phishing (1)** and **Business Email Compromise (6)** land well with general staff audiences — the entry vector (a phishing email, a compromised executive inbox) is something everyone recognises.
- **Supply Chain Compromise (2)** and **Cloud Misconfiguration Exploitation (4)** are good choices for technical or DevOps-adjacent audiences — they hinge on vulnerability management and compliance guardrails rather than human behaviour.
- **DDoS Attack with Extortion (7)** is mitigated by a single control (MDR), making it the simplest playbook to use as a first demonstration — one toggle, one clear before/after contrast.
- **State-Sponsored APT (8)** and **Insider Threat Data Theft (3)** work well for more senior or security-literate audiences, since the narrative requires more context to land.
- **IoT / OT Environment Breach (9)** is worth including for any audience with operational technology or facilities-adjacent assets in scope.

5. Choosing a Topology Tier

Tier	Nodes	Best for
Simple Network	7	First-time players, short sessions, or explaining the core mechanic without visual clutter.
Medium Complexity	17	The default and best general-purpose choice — enough structure to feel realistic without overwhelming a first session.
Complex	27	Technical audiences, or once a group is comfortable with the mechanics and ready to reason about a full enterprise estate.

Tier	Nodes	Best for
Complex Multi-Site	39	Advanced sessions specifically discussing multi-site risk, WAN links as attack surface, or DR/business-continuity scenarios.

6. Compliance Frameworks Referenced

Every SIEM log entry that references a compliance outcome cites one of five frameworks. These are real (or, in the case of the Cyber Resilience Bill, real-styled) frameworks — use them as a bridge into a conversation about your own organisation's actual obligations.

Framework	What it represents
NCSC CAF	UK NCSC Cyber Assessment Framework — outcome-based guidance for essential-service organisations.
Cyber Essentials Plus	UK government-backed technical certification for basic cyber hygiene controls.
Cyber Resilience Bill (CRB)	Placeholder framework used in-app for mandatory incident reporting and governance obligations, styled on the EU Cyber Resilience Act.
NIST CSF 2.0	US NIST Cybersecurity Framework — Identify, Protect, Detect, Respond, Recover.
SOC 2 Type II	AICPA Trust Services Criteria — audited controls for security, availability, and confidentiality.

7. Facilitation Tips & Discussion Questions

General tips

- Let the group make the call together on at least one phase choice per playbook, rather than you clicking through alone — the discussion in the moment of deciding is where the learning happens.
- Deliberately run one playbook to a full breach early on. Groups remember the cascade far more vividly than a clean containment.
- Don't frame asset-response actions as a "correct answer" quiz — the point is that Isolate is usually strongest and Take No Action is usually weakest, but context (what the asset is, what it costs to disrupt it) should be part of the conversation.
- If a group over-isolates every asset "just to be safe", use it as a prompt: what would happen to the business if every one of these assets actually went offline at once?

Discussion questions

- Which single control, if it had been switched on, would have changed today's outcome the most — and why wasn't it on by default?
- When you isolated that asset, what would that actually have meant for the people who depend on it during working hours?
- Why might sending external comms before internal comms be a mistake, even if the information in both messages is identical?
- The Session Debrief said your regulatory notification was late. What's the real-world deadline your organisation would actually be working against?
- If you re-ran this exact playbook with the same topology, would you make the same choices again? What would you change first?

8. Known Limitations

- This is a training aid, not a risk model — the probabilities behind asset actions are illustrative, not derived from real incident data.
- Nothing persists. There is no save state, no scoring history across sessions, and no way to export a transcript beyond screenshotting the SIEM log before it's reset.
- Each playbook always targets the same six core assets regardless of topology tier — the additional nodes in larger tiers add visual and contextual richness but are not individually targetable by the scripted playbooks (though they are still fully clickable for asset-response practice).
- The tool runs entirely client-side in the browser; closing the tab or refreshing the page ends the session exactly like pressing Reset would.

9. Troubleshooting

Symptom	Likely cause / fix
Playbook Engine stays locked	Start Simulator has not been pressed, or the countdown is still running — wait for "SIMULATION LIVE" to clear.
Clicking a node does nothing	The simulator has not been started yet. Node actions are only available once it is live.

Symptom	Likely cause / fix
Topology looks tiny or clipped	The larger tiers are wider than the visible canvas by design — scroll within the topology panel, don't resize the browser to compensate.
Docker container won't start	Confirm port 8080 is free on the host, or change the left-hand side of the ports mapping in docker-compose.yml.
Styling looks broken / unstyled	The app loads Tailwind CSS from a CDN at runtime — confirm the machine running it has outbound internet access, even when served locally via Docker.