

● STARTING SOON • 19:00 LONDON / 21:00 KYIV

The new AWS Lambda for AI agents

OLEH LEVCHENKO • SENIOR FULL STACK DEVELOPER

The new AWS Lambda for AI agents



Oleh Levchenko

SPEAKER

Oleh Levchenko

Senior Full Stack Developer

- 9+ years of experience
- JS / TS · React · NestJS · AWS
- Founder of IT Friday

linkedin.com/in/olevchenko youtube.com/@zloyleva
itfriday.community

Today

01 • DAY 2 • TALKS

- **Julian Wood**, AWS — Serverless Compute for Workloads You Didn't Expect
- **Kyra Wulffert**, Databricks — From Shipping Models to Operating Agents
- **Grace Briody**, Microsoft — The Friction Tax
- **FinTech Wales panel** — AI in Financial Services
- ~~Ian Massingham~~, Anthropic — ~~Going AI Native~~ cancelled
- **Matt Johnson & Charmion Akande**, UBDS — Digital Sovereignty
- **Gunnar Grosch**, Sinch — Building Conversational Agents in Code...
- **Tom Brewer**, DVLA — Modern Ambition, Legacy Reality
- **Samuel Waymouth**, AWS — AI Governance
- **Jim Stamp**, Made Tech — Pairing Internal Experiments...
- **Robert Kitching**, PureCyber — Impact of AI on the SOC

02 • DAY 1 • WORKSHOPS

- **AWS Lambda MicroVMs** — Michael Hume, AWS
- Code sandbox · coding agent on PRs · ephemeral CI runner · multi-tenant SaaS
- ~~Anthropic Workshop~~ cancelled

03 • MY TAKEAWAYS

- The human-in-the-loop paradox
- Runbooks written for human attackers
- What I'll do differently at meetups



THE EVENT • 20-21 OCT 2026 • DEPOT, CARDIFF

ServerlessDays Cardiff

Part of ServerlessDays — a non-profit, community-run series: 57 events in 48 cities, Lviv included

6th

edition, every year

2 days

workshops, then talks

~250

people in the room —
and all of them fed

£35

ticket — workshop day
included

10 talks

plus the first-ever panel

£0

to speakers — all came
at their own cost

cardiff.serverlessdays.io youtube.com/@ServerlessDays

linkedin.com/company/serverless-days-cardiff

01

Serverless Compute for Workloads You Didn't Expect

Julian Wood • Principal Developer Advocate, AWS Serverless • opening keynote



LAMBDA TODAY

10 years of "a function handles an event"

15T+

Lambda invocations every month

1.8M+

customers

2014

launch: 1 request per environment, 15 min max, pay for the whole run

**“We’re bringing a bit of the servers
back into serverless.”**

Julian Wood, on Lambda web functions

AWS Lambda form factors

FORM FACTOR

WHAT IT IS

Event functions	The Lambda you know, renamed. SQS, Kinesis, S3 triggers. One request per environment
------------------------	--

Durable functions	Workflows in code: checkpoints and replay. Pause up to a year without paying
--------------------------	--

Web functions	A standard HTTP server. Express, Fastify, Hono — no adapters
----------------------	--

MicroVMs	A VM sandbox you control. Own kernel, starts from a snapshot in < 1 s
-----------------	---

Where it runs: **on-demand** or **Lambda Managed Instances** — EC2 in your account, EC2 pricing

Web functions: Lambda listens on a port

- A normal HTTP server: Express, Fastify, Hono — **no adapters**, `npm start` locally = the same in the cloud
- One environment serves many requests at once: **64 by default, up to 128**
- **Active-CPU billing** — waiting on a DB or an LLM costs little. Memory and requests billed separately
- Streaming responses, multi-Region endpoint in one command
- Today: **Node.js 24 on Arm64**, 2 vCPU / 2 GB per environment

Durable functions: workflows as plain code

- Checkpoints like a **save game** — after a crash, replay skips the finished steps
- `ctx.step` · `map` · `parallel` · `callback`
- Pause up to **a year** and pay nothing — e.g. waiting for a manager's approval
- Idempotency built in
- Node, Python, Java, C# — Go and Rust next

LIVE DEMO • SSE DASHBOARD ON WEB FUNCTIONS

300 streams, up to 200 open at once

5

environments held it all — peak 59 streams on one

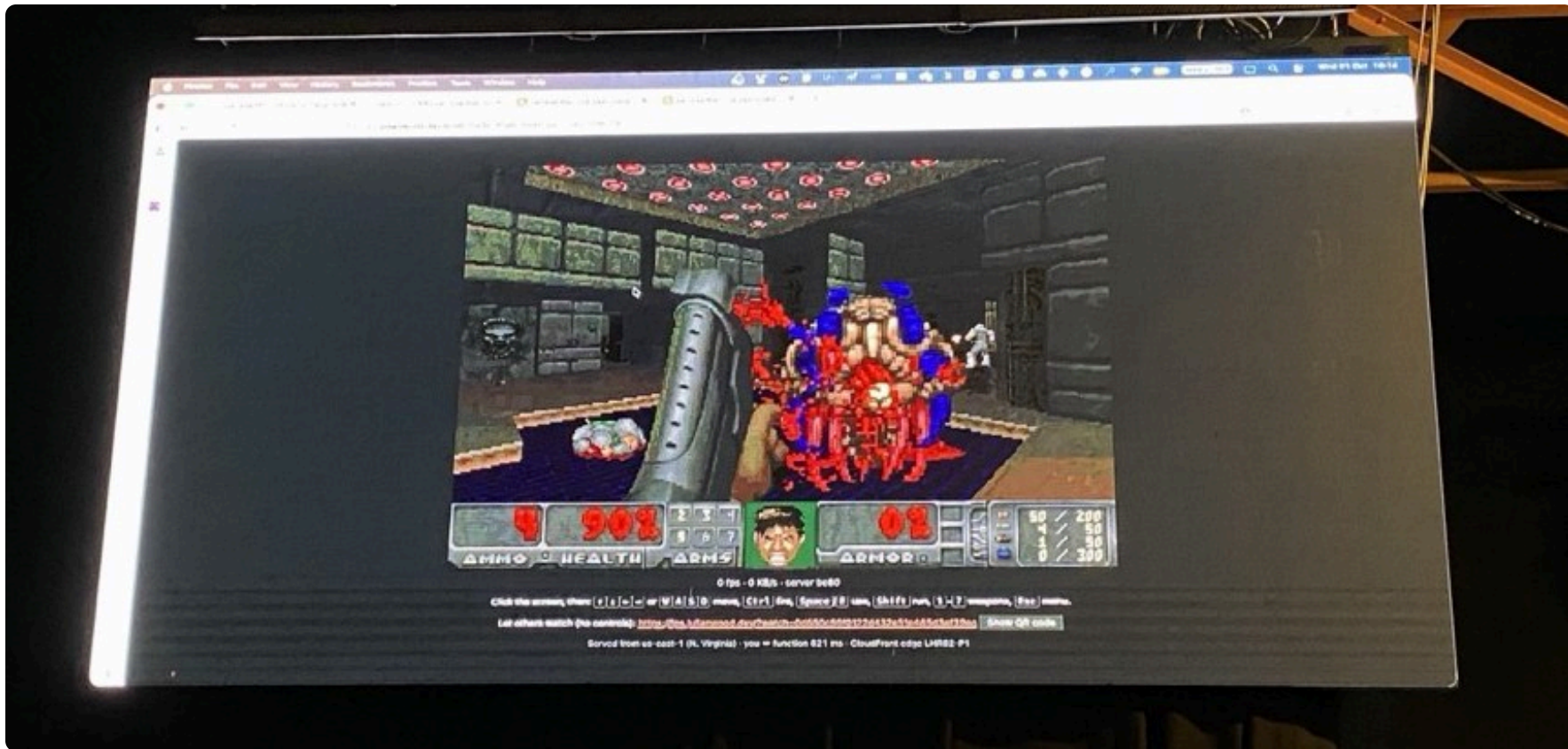
4.75 s

of CPU for 484 requests, with streams open up to 90 s

90 → 16

ms from the UK: us-east-1 vs the global endpoint (London)

Event functions: one open stream = one environment → up to 200



And of course: Doom running in Lambda – built with Kiro and Claude

Oleh Levchenko

02

From Shipping Models to Operating Agents

Kyra Wulffert • AI Specialist Architect,
Databricks • co-author of The Big Book of
AgentOps



THE GAP IS OPERATIONS

Agents are everywhere in pilots — and rare in production

97%

of executives say they are
deploying agents

12%

have anything running at
production scale

89%

have observability on their
agents

52%

actually run evaluations

Numbers from Kyra's slide

Four levels of complexity

Most teams climb the architecture ladder faster than the controls ladder

- 01 **LLM + prompt** — one call, prompt in, text out

- 02 **Agentic retrieval** — the model decides what to retrieve and when

- 03 **Tool-calling agent** — plans, calls tools, acts on real systems

- 04 **Multi-agent** — a supervisor routes to specialists

One request, two views

The log

```
user="Why was I charged twice?"  
answer="Your account shows one  
  charge of £42.00 on 3 June."  
latency=6.8s  status=200
```

200 OK. Confident. Well-formatted.
Wrong.

The trace

```
agent supervisor  
  llm  classify_intent  
agent billing_agent  
  tool get_invoices  TIMEOUT ×2  
  tool get_invoices  CACHE-STALE  
  llm  compose_answer  
judge groundedness  0.31
```

Evals: from cheap to expensive

- 01 **Deterministic checks** on every request — format, PII, rules

- 02 **LLM judges** on a golden dataset in CI + a sample of production

- 03 **Expert (SME) review** — a few times a year

- 04 **User** 👍 / 👎

Golden dataset comes from traces: vibe checks → traces → dataset → **CI gate**

Common anti-patterns

ANTI-PATTERN

FIX

Starting too broad

One use case, one measurable metric, then expand

Over-engineering

Simplest pattern first — direct calls and tools before multi-agent

No human in the loop

Approval for high-stakes actions: refunds, deletes, payments

No versioning

Every prompt, tool and route is a versioned artifact

The biggest blocker isn't technical: agree on roles and KPIs up front.

C-level wants ROI, PMs want adoption, engineers want speed, data scientists want their metrics

03

The Friction Tax

What's Actually Slowing Developers Down
in the Age of AI? • Grace Briody, Solutions
Engineer for Cloud and AI Apps, Microsoft



Developers got a Lamborghini — **and they're stuck on the M25.**

Grace's analogy: AI gave us a faster tool, but everything around it hasn't changed

DEVELOPERS DO MORE THAN WRITE CODE

Nine stages of searching, waiting and approving. One stage of writing code

32%

of developers' time goes to writing or improving code

90%

of developers lose 6+ hours a week to non-coding friction

Sources on Grace's slide: Tidelift / The New Stack developer survey 2019 • Atlassian State of DevEx 2025

DORA • AI AMPLIFIES WHAT'S ALREADY THERE

A faster step doesn't create a faster system

+25%

AI adoption (DORA 2024)

-1.5%

delivery throughput

-7.2%

delivery stability

2025: throughput turned positive — stability did not

Drag, Fog, Heat and Wall

FRICTION WHAT IT MEANS

EXAMPLE

Drag	It takes longer than it should	Waiting for access, an environment
Fog	The correct path is unclear	Three docs, Stack Overflow, and AI says a fourth thing
Heat	The experience is eroding trust	"Everyone knows deploying this is hard" — people leave
Wall	The developer cannot continue	No access, security review, approval queue

Five places AI removes friction

WHERE	HOW	REMOVES
Design reviews	Pattern check before the architecture board	Drag, Fog
Environments on demand	Plain request → approved IaC → auto-teardown	Wall, Drag
Generated tests	Failure paths by default — review coverage, not syntax	Drag
Operational visibility	Alert → logs + traces + recent changes → hypothesis	Fog, Heat
Repetitive work	Dependency bumps, changelogs, release notes	Drag

None of them is writing code. A human gate stays in every one

Three principles for reducing the friction tax

- 01 **Look beyond coding** — timebox a week, log every wait, search and retry. That's your friction backlog

- 02 **Don't automate friction** — before building an agent around a process, try deleting the step

- 03 **Measure outcomes, not AI usage** — lead time and change failure rate, not prompts or lines of code

“We don’t need 10x developers. We need development environments with 10x less friction.”

Grace Briody

04

AI in Financial Services

Innovating Without Compromising Trust ·
FinTech Wales panel, hosted by Sarah
Jones (CEO) · the first-ever panel at
ServerlessDays Cardiff



REAL NUMBERS FROM A BANK

How many AI projects actually reach production?

~10%

of proofs of concept — people expected 50%

~50%

of pilots — people expected 90%

Fix: rank AI initiatives by value × chance of reaching production, and take that to the board

Three takes from the panel

- **Human in the loop as a backstop is “performance art”** — we default to it because we lack conviction in our tools
- **Tokens feel like “buying chips to go to the casino”** — price by human hours saved, not by tokens
- **Hairdressing is “really all about therapy”** — keep AI away from the human touch: relationships, creativity, accountability

05

Digital Sovereignty

From theory to implementation · Matt Johnson, CTO, UBDS Group & Charmion Akande, Data Platform Engineer, UBDS Digital



Would your services keep running if access to the cloud changed?

Not “is it right or wrong” — just how it works in practice

Four events. Four ways access can fail

WHEN	WHAT HAPPENED	THE LESSON
Feb 2022 · Ukraine	Government data moved to the cloud after off-site hosting rules were relaxed	Physical location became a liability
Mar 2026 · UAE, Bahrain	AWS facilities hit by drones	A cloud region is still physical
Jun 2026 · US	Export directive: Claude Fable 5 access suspended globally, restored 1 July	Policy can interrupt APIs overnight
Oct 2026 · Russia	Yandex reports attacks on its Sasovo and Kaluga data centres	Domestic ownership is no force field

As presented by the speakers

🇺🇦 UKRAINE, 2022 — CASE

80 metres from losing every system of record

~11 days

before the invasion, Ukraine lifted the ban on keeping government data outside its borders

Months

AWS, Microsoft and others spent helping move critical data to the cloud

80 m

from the main government data centre — where a bomb landed

What sovereignty actually means

- **"In-country" is not a resilience strategy** — your own servers are great until a bomb lands on them
- **Multi-region is not sovereign** — AWS regions are isolated, but the global partition is run from us-east-1
- **Moving the code is the easy part** — data, DNS, identity, keys, CI/CD are not. *A plan is not an option until you've exercised it*
- **Proportional to criticality** — out of hundreds of services, maybe a dozen truly matter

Eight dimensions of dependency

The unit of analysis is a critical service and all its dependencies

-
- **Technology** — platforms, APIs, runtimes, proprietary services
-
- **Data** — portability, gravity, lineage, rehydration
-
- **Operations** — monitoring, backup, tooling, support
-
- **Security** — identity, access, key custody, assurance
-
- **People** — skills, knowledge, supplier expertise
-
- **Commercial** — licensing, commitments, switching costs
-
- **Contracts** — exit rights, termination, transition support
-
- **Jurisdiction** — ownership, law, supply chain, policy

AWS European Sovereign Cloud

- **A separate partition, not a region** — its own IAM, control plane, endpoints and KMS. “Single-vendor multi-cloud”
- No VPC peering or Direct Connect across the boundary — **only VPN or HTTPS** with separate credentials
- **Demo:** copy S3 data from commercial AWS over HTTPS → switch commercial off → the app in ESC keeps running
- Still open: where DNS lives, where IaC and pipelines run, who owns identity, how to monitor across the boundary

06

Building Conversational Agents in Code

with AWS Lambda Durable Functions •
Gunnar Grosch, Developer Advocate, Sinch
• AWS Serverless Hero



WHY CONVERSATIONS ARE HARD

Conversations run on human time

15 min

a Lambda invocation ends. The customer answers tomorrow

20%

ten steps at 85% each succeed end to end. Redoing finished steps makes it worse

3 channels

SMS today, WhatsApp tomorrow, a phone call when it gets urgent

The usual fix spreads the conversation across queues, a state table and polling

A conversation is a loop

```
export const handler = withDurableExecution(  
  async (event, context) => {  
    for (let turn = 1; turn ≤ MAX_TURNS; turn++) {  
      const reply = await context.waitForCallback(  
        `reply-${turn}`, notify, { timeout: { hours: 24 } })  
  
      const answer = await context.step(`answer-${turn}`,  
        () => agent.invoke(reply))  
  
      await context.step(`send-reply-${turn}`,  
        () => sendSms(event.from, answer.text))  
    }  
  })
```

`waitForCallback` sleeps, nothing is billed • `step` runs once and checkpoints the result

The handler will replay

- On replay the code runs **from the top** — model calls, IDs, timestamps and sends go **inside step**
- **Step names are a contract** — rename them mid-execution and replay won't match
- **step $\neq$ exactly once** — sends still need an idempotency key
- An execution is pinned to a function version — publish versions, deploy via aliases

Two jobs: recover and remember

	DURABLE EXECUTION STATE	AGENT STORAGE
Holds	Which operations ran and their results	What the customer said and what was decided
Scope	One execution of one function	A customer, across executions, channels and agents
Lifetime	The execution, then 1–90 days	As long as you decide
Read by	The runtime on replay	Any agent: SMS, WhatsApp, voice

Durable state is for recovery. Agent storage is for context. You need both

Green history, **wrong reply.**

The workflow never failed. The model sometimes did — so check the result against the booking system before you send

07

Modern Ambition, Legacy Reality

Transforming through the constraints of
yesterday · Tom Brewer, Chief Digital and
Technology Officer, DVLA



Six years off the mainframe

DVLA in numbers

- **48M** vehicles · **53M** drivers
- **£9.9B** vehicle tax a year (forecast)
- **118M** letters posted a year
- **85.8%** digital take-up

The story

- **Customer-facing first** — one account on GOV.UK One Login
- **Mainframe → AWS** — codebase -25%, no more manual tests or deploys
- **Medical declarations** — digital take-up 16% → 30%+ in under six months
- **Then the punch** — pay-per-mile EV tax by 2028. *“Everybody’s got a plan until they get punched in the face”*

08

AI Governance: From Foundations to Agentic Speed

Samuel Waymouth • Senior Specialist
Solutions Architect, Security & Compliance,
AWS



THE GAP BETWEEN ADOPTION AND GOVERNANCE

AI rarely makes it to production

88%

of AI agent pilots never reach production

80%+

of AI projects fail — twice the rate of non-AI IT

42%

abandoned most AI initiatives in 2025 (17% in 2024)

From Samuel's slide · Forrester/IDC, RAND, S&P Global

**Agents work at machine speed.
Governance still works in tickets and
quarterly reviews**

Three parts, controls in between

The AI Security Triad

- **Inference** — the model. Mostly the vendor's terms
- **Agents** — just code in a repo. Scan it like any code
- **Tools** — MCP servers, APIs, A2A

Controls at the interfaces

- **Network** — isolation, no egress
- **AuthN / AuthZ** — least privilege
- **Guardrails** — content checks
- **Policies** — allow / deny boundaries

Earned autonomy

STAGE	WHAT THE AGENT GETS
Development	Unrestricted experiments, no production data
Staging	Policy in LOG_ONLY — observe without blocking
Pre-production	Evaluations, LLM as a judge
Production	Policy in ENFORCE — deterministic controls

Score drops below the threshold → auto-disable → investigate

Human in the loop **doesn't scale.**

Too many tickets in the queue — and the human approves everything to make them go away

Samuel's takeaways

- 01 Define your scope and start with a proper foundation
- 02 Set up visibility — you cannot govern what you cannot see
- 03 Centralise policy enforcement, decentralise development
- 04 Use every control your risk posture needs — and keep monitoring

Startup idea from the stage: a governance agent that updates the risk register on every commit

09

Pairing Internal Experiments with Public Sector Impact

Jim Stamp · Head of Technology, Made Tech · stood in for Sam Tripp



Coding is no longer the bottleneck

Ideation › Research › Design › **Development** › QA › Deployment

The hype and tooling went to development. The constrained roles have the longest queue and the least investment

EVERY ROLE GETS AGENTS

AI First, not just for engineers

16

job families given agentic capability

30+

verified agents and skills — and growing

10

testable prototypes in hours — the team estimated two weeks

Their own caveat: an estimate, not a measurement · one instance · prototypes, not production code

Agents for all — at a very different distance

An engineer's Monday

- Work arrives as a ticket
- They open the repo
- **The agent is already in VS Code**

A researcher's Monday

- Work arrives in a doc, a board, an invite
- The agent is somewhere else
- **Reaching for it means leaving the work**

Three things you can do tomorrow

- 01 Find your constraint before you buy a tool for it
- 02 Put the agent where non-engineers' work arrives
- 03 For every skill you share, say where it has been tested

Made Tech's AI risk assessment framework: madetech.github.io/ai-risk-assessment-framework

10

Impact of AI on the SOC

Contrasting both sides of AI for attack and defence · Robert Kitching, CTO, PureCyber · no slides, last talk of the day



Three alerts, one attack

WHEN

SIGNAL

00:00

Sign-in from an unusual country

+10 min

A device never seen before

+20 min

5,000 emails read or exported

Alone, each means nothing. Together — an indicator of attack. AI helps find the pattern

Same intent, more attempts per day

Attackers

- Phishing with better spelling
- Recon in minutes: who are the admins?
- Malware re-signed faster than AV
- Deepfake voices on the help desk

Defenders

- Cut the noise — most alerts are benign
- Detection rules as code, in plain text
- Translate rules: Sentinel ↔ Splunk
- Same report, whoever the analyst

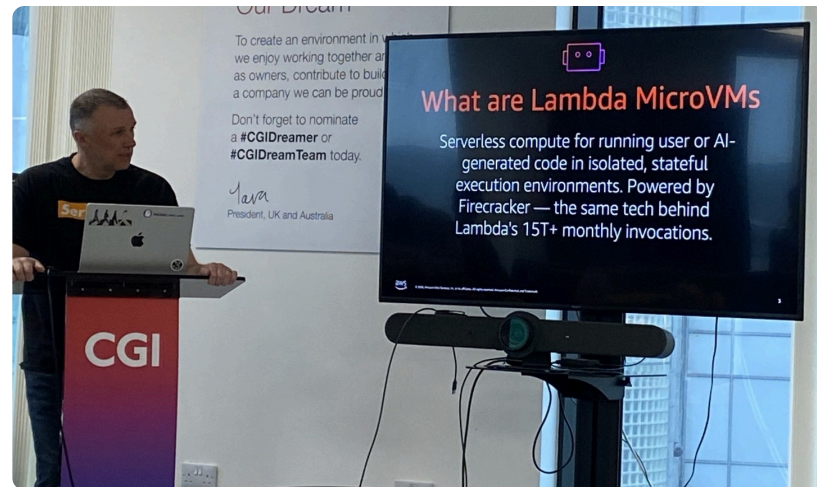
AI advises. It doesn't act — yet.

"I checked this IP, it's fine." Checked with what? Data from six months ago?
And the playbooks were written for humans, not agents

11

AWS Lambda MicroVMs

Day 1 workshop • Mike Hume, Pete Davis,
Julian Wood • AWS • hosted at CGI, Cardiff



Functions vs MicroVMs

	FUNCTIONS	MICROVMS
Model	Event-driven handler	Any app on ports
Lifetime	Up to 15 min	Up to 8 hours
State	Stateless	Stateful suspend / resume
Scaling	Auto-scaling	Dev-controlled

Anti-pattern: don't put a MicroVM on a queue — 1000-rps event traffic is a function's job

AT A GLANCE

A whole VM per session

8 h

maximum lifetime

32 GB

memory and 32 GB disk

16 vCPU

with 4× burst

ARM64 Graviton · Firecracker · no shared kernel · per-second billing

Three building blocks

01 **Image from a Dockerfile** — AWS builds it and snapshots the running app

02 **Lifecycle hooks** — `/ready` `/run` `/suspend` `/resume` `/terminate`

03 **Idle policy** — idle → suspend → terminate, auto-resume on a request

Every MicroVM gets its own HTTPS endpoint and a short-lived auth token

What we built

MODULE	WHAT HAPPENS
Code sandbox	POST Python code → it runs in an isolated VM
Coding agent	A PR → a VM with Claude Code reviews the diff
CI runner	Every push → a clean VM runs the pipeline, then disappears
Multi-tenant SaaS	One VM per customer behind a control plane

The human-in-the-loop paradox

Almost every speaker said: keep a human in the loop. The market is firing, not hiring

- Who is that human in five years, if nobody hires juniors today?
- One senior reviews ten agents — review becomes the bottleneck, then a formality
- Fair counterweight: 2021 overhiring and interest rates. AI is often a convenient excuse

Security runbooks assume a human attacker

- **Speed** — hours for a human, minutes for AI. Thresholds are tuned to human pace
- **Scale** — personalised phishing to thousands at once
- **Insiders** — our own agents with access. A prompt in a PR diff can steer a review agent
- **Logs** — often can't tell an agent's action from a human's

Automated attack → the response has to be partly automated too

What I'll do differently at meetups

- 01 **Network with the speakers** — I used to focus on the audience

- 02 **Go small** — in Cardiff speakers are reachable, less pomp than London

- 03 **Prepare for workshops** — the steps are copy-paste. Bring questions on the topic and around it, spend the time talking

QUESTION TO THE CHAT ↓

Is your company still hiring juniors — or betting on agents?

Thanks for watching

See you next Friday • 19:00 London / 21:00 Kyiv



Telegram

announcements + chat



Discord

tech talk between streams

itfriday.community

all streams & speakers