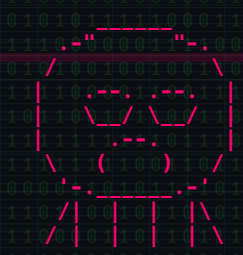
// 04/2026 // AIR-GAPPED EDITION

$ cat /dev/mag | hexdump -C

PLANET EXPRESS

FINDING PATTERNS IN THE NOIZE

EAT SLEEP HACK REPEAT



```
00004000  f1 d7 94 07 5b ed 01 15 22 2c c1 80 46 f1 ab a
00004010  99 8d f1 95 e7 04 4b 19 bc ab 42 f6 ce 65 f7 6
00004020  92 2e 81 f9 c2 33 22 6b 70 6b bb 0c 86 fd 67 7
00004030  a4 37 2a 6f 1c 60 d7 bc b9 66 a8 ed b2 04 d7 5
00004040  e6 d0 b4 3e ba 9d c5 bd f9 da dc ef ba 05 68 8
00004050  6d 8e d6 f1 db 6d 82 e1 0e 94 79 dd 9f 44 81 d
00004060  61 c0 de ca 6a 43 21 69 04 cd 83 8a 32 38 81 2
00004070  3e 77 26 2a bd de 61 6b 35 4f 27 7f 84 dd c7 1
00004080  ed 2b 2c 57 dd 09 70 e8 1b 8d b7 4b e2 77 2e e
00004090  90 a5 a8 90 2b 7c 53 2f 66 ef 20 f1 ea ef 94 c
000040a0  7f 08 09 04 56 09 04 72 85 12 ab 23 75 3c 2b 6
000040b0  62 24 ae 85 2d 99 29 c0 b9 d8 a6 23 b8 7b 6f 3
000040c0  4e 1b 1a b9 61 2b 0c b0 5f 1f cf a6 42 34 c7 0
000040d0  b4 28 c2 d8 6a e6 ba 9e a5 da f0 d7 88 fb 47 a
000040e0  af a2 93 6a 95 78 5c 7a 04 b1 87 fd f3 c8 dc c
000040f0  1f 45 52 7f c7 89 13 f0 54 17 3e 31 a5 8d 43 2
00004100  40 ab ce ef c9 a8 0d 95 64 44 b1 28 e5 fd ad d
00004110  2e bb 79 77 c5 ae 32 9c 07 b5 13 25 12 f3 40 b
00004120  a2 68 3b 23 0d bc 86 85 90 50 32 53 b2 3e 31 1
00004130  73 53 d0 42 ce 89 19 a0 b0 84 5b a3 1e 40 1b 6
00004140  61 57 00 0f 15 98 7f 2b ad 81 e3 cf ae bc aa 9
00004150  49 53 33 b4 22 5b 65 ba 21 79 1e 33 4f 8c c3 4
```

free as in beer

// FEATURE 01 // SUPPLY CHAIN

THE AXIOS AFFAIR

On March 31, 2026, axios 1.14.1 and 0.30.4 were malicious for three hours. 180 million downloads a week. Here is how it ran.

There is a specific shape of modern dependency: small, boring, solves exactly one problem, and has installed itself into roughly every JavaScript project on Earth. axios is that dependency. Around 180 million npm downloads per week, split across two main branches: 1.x and 0.30.x. The HTTP client most teams did not choose, running inside their build anyway.

Ubiquity is the risk. What makes a supply-chain attack work is not cleverness — it is reach.

The history has precedent. event-stream in 2018 reached two million weekly downloads before its maintainer handed it to a stranger, who shipped a wallet stealer aimed at Copay users. ua-parser-js in 2021 carried a coin miner and credential stealer through a compromised maintainer account. colors.js and faker.js in 2022 were sabotaged by their own author. node-ipc in 2022 wiped files based on geolocation. xz-utils in 2025 was a patient three-year campaign that came within one beta release of backdooring sshd on every Linux distribution.

On March 31, 2026 at 00:21 UTC, axios itself was hit. Three hours later, at 03:20, the window closed again. What follows is the mechanics of the attack, stage by stage, as it actually ran.

```
$ npm install --production

added 1847 packages in 23s

19 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

$ node server.js
server listening on :3000
POST packages.npm.org/product1  200
POST packages.npm.org/product1  200
POST packages.npm.org/product1  200
POST packages.npm.org/product1  200

# found 0 vulnerabilities
# found 1 apocalypse
```

```
[0] maintainer account hijacked
     └─ new email: ifstap@proton.me
          ▼
[1] plain-crypto-js @ 4.2.0 / 4.2.1
     └─ injected as axios dependency
          ▼
[2] postinstall: node setup.js
          ▼
[3] SILKBELL (XOR + Base64 dropper)
     ├─ win: %PROGRAMDATA%\wt.exe
     ├─ mac: /Library/Caches/...mond
     └─ lin: /tmp/ld.py
          ▼
[4] WAVESHAPER.V2 → sfrclak.com:8000
          ▼
[5] persistence: Run\MicrosoftUpdate

2026-03-31 · 00:21 - 03:20 UTC
```

// FEATURE 01 // SUPPLY CHAIN // CONT.

Stage zero: compromised maintainer account. Not a typosquat, not a side-channel — a legitimate axios maintainer lost control of their npmjs account. The first visible artefact was the new contact email on the account: ifstap@proton.me.

Stage one: a hidden dependency. The attacker added a new package named plain-crypto-js to axios, published in versions 4.2.0 and 4.2.1. The name sounded harmless. The whole attack lived inside that dependency, not in axios itself — a diff against axios would have shown nothing unusual.

Stage two: postinstall hook. plain-crypto-js carried a scripts entry in its package.json that ran `node setup.js` on every npm install. No user consent, no prompt, just code executing at install time.

Stage three: SILKBELL, the dropper. setup.js was obfuscated with XOR and Base64; C2 URLs and OS-specific commands were only assembled at runtime. fs, os and execSync were required dynamically to dodge static analysis. After the drop, setup.js deleted itself and renamed package.json to package.md to cover its tracks.

Stage four: OS-specific payload. On Windows, SILKBELL copied powershell.exe to %PROGRAMDATA%\wt.exe — the filename of the legitimate Windows Terminal, used as camouflage. A PowerShell script was pulled via curl from packages.npm.org (a lookalike domain, not the real npm registry), POST body `product1`. On macOS, a Mach-O landed in /Library/Caches/com.apple.act.mond, body `product0`. On Linux, a Python backdoor went to /tmp/ld.py, body `product2`.

Stage five: WAVESHAPER.V2, the actual backdoor. It beaconed back to sfrclak.com (142.11.206.73) over port 8000 every 60 seconds. Commands: `kill`, `rundir` (directory enumeration), `runscript` (AppleScript), `peinject` (PE injection into a process). One hard-coded User-Agent stayed put: mozilla/4.0 (compatible; msie 8.0; windows nt 5.1; trident/4.0) — an IE8 masquerade the WAVESHAPER family has been carrying for years, and the thing that ultimately made attribution easy.

Stage six: persistence, Windows only. A hidden %PROGRAMDATA%\system.bat plus a registry entry under HKCU\Software\Microsoft\Windows\CurrentVersion\Run with the value "MicrosoftUpdate" brought the backdoor back on every user logon. No persistence was documented for macOS or Linux — probably because the one-shot reach was already enough.

Stage seven: aftermath. Hundreds of thousands of stolen secrets may be circulating as a direct result. The defences are not exotic — lockfiles, `npm ci --ignore-scripts`, scoped tokens, egress allow-lists in CI, ephemeral runners. None of it is deployed at the scale the ecosystem needs. It has been that way since 2012, and will be that way in 2030. The only open question is which package is next.

// FEATURE 02 // FIELD REPORT

SPEEDRUNNING TRYHACKME ANY %

A bot that works through an entire cybersec syllabus on its own — and gets a little sharper with every room.

TryHackMe is a learning platform for IT security. You get assigned a "room" — an exercise with explanatory text, questions, and usually a virtual target machine. Correct answers earn XP, fill the progress bar, unlock the next room. Hundreds of them, sorted into learning paths.

The question behind this experiment: what happens when you point claude-code — an LLM agent that lives in the shell, reads and writes files, runs commands — at that syllabus and just wait?

The first piece is trivial. A Python script starts Playwright, does a one-time manual login (the captcha wants a human), saves the cookie jar. From then on, a small tool talks to the internal API directly: GET /api/v2/rooms/tasks returns every question in a room, POST /api/v2/rooms/answer submits an answer. That is the entire platform from the inside.

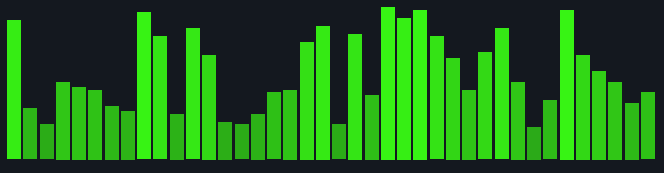
The agent gets a room, reads the task, figures out the answer, types it in. As long as the question is "which word appears in the second paragraph", this is trivial. The moment nmap has to be run against a virtual machine or a login form has to be cracked, the API-clicker turns into a small hacker: it opens a shell, calls nmap 10.10.x.y, reads the open ports, picks the next tool on its own.

An orchestrator doesn't run one agent but many in parallel. Each gets exactly one room, a fixed time budget, its own context window. While one is fighting SQL injection, the second is clearing Linux forensics, the third is solving a reverse-engineering puzzle.

 # AGENT_POOL.dispatch
path: cybersecurity101

agent-001	**nmaplivehostdisc**	18:42	DONE	+400
agent-002	**linuxforensics**	—:—	RUN	
agent-003	**webfundamentals**	11:29	DONE	+300
agent-004	**introtodfir**	—:—	RUN	
agent-005	**burpsuitebasics**	03:11	FAIL	
agent-006	**wiresharkbasics**	—:—	RUN	
…	…	…	…	…

pool size	rolling
completed	34 / 43
wall runtime	07:21:44
tokens burnt	2.8M
recipes	growing



rooms cleared · rolling 24h · y=count, x=hour

// FEATURE 02 // FIELD REPORT // CONT.

The interesting part is what happens between rooms.

Next to the code sits a directory called skills/ — the bot's memory. Inside: a SKILL.md describing how to approach any room (read first, then scan, then exploit on purpose). A lessons.md collecting observations from prior solves ("some static-site flags are base64-encoded inside the JS bundle; grep is enough"). A recipes folder with one markdown file per attack technique — SMB recon, JWT tampering, buffer overflow — written as a ready-to-use playbook.

Before each solve the agent reads all of it. After each solve, if it has seen something new on the way, it writes the finding back. New file, new entry, small diff.

The bot grows with the corpus. Rooms that broke it in the first week fall in minutes in the second, because the matching recipe appeared in between. The setup tunes itself along the way — every solved room leaves a little more documentation for the next one.

Two things happen in parallel here, and they refuse to separate cleanly. Learning: about APIs, captchas, rate limits, the anatomy of an attack. Cheating: agents clicking "Check" on questions nobody read. The progress bar moves anyway.

For the CTF scene this says little that is new. A leaderboard was never a credential, just an attendance signal. Automation was going to catch up eventually; it just got there sooner than the curricula budgeted for.

For IT security as a whole it says more. Attackers have always worked with scripts; what is new is how much of the work an LLM bot with a few markdown files as memory will now take off their hands. The threshold from "experienced red teamer" to "someone with a free weekend" has collapsed. The honest question for anyone on the blue side is which of those things your counterpart can now automate that you cannot. The answer tends to sting.

// TOOL SPOTLIGHT

NMAP

The Swiss army knife that refuses to retire.

Gordon Lyon — Fyodor — released nmap in 1997 as an article for Phrack. Twenty-nine years later it is still the first thing anyone types when a new network appears in front of them. There is no higher compliment you can pay a piece of software.

What makes nmap durable is that it is not opinionated. It does not try to be pretty. It does not try to be an agent. It does not want to live in your browser. It is a flashlight you aim at a network, and it tells you what it saw.

The most underrated thing about nmap is that the naked form is already enough. Just `nmap <target>`. No flags, no scripts, no tuning. It scans the top thousand TCP ports, guesses what is running on each, and hands back a shape of the target. That shape is usually enough to decide where to look next. The cheatsheet below exists to answer questions the first scan raised.

CHEATSHEET

`-sS`	stealth SYN scan
`-sV`	probe version info
`-O`	OS fingerprint
`-A`	aggressive (all the above)
`-p-`	all 65535 ports
`-T4`	fast timing template
`--open`	show only open
`--script`	NSE script engine
`vuln`	vuln detection scripts
`smb-enum-*`	smb recon pack
`http-title`	grab http titles
`-iL <file>`	read targets from file

$ nmap without flags:

```
nmap 10.10.10.42
```

also valid: nmap example.com nmap 192.168.1.1-254 nmap 10.0.0.0/24
top 1000 TCP ports, service guesses.

// COLUMN // TERMINAL TRICKS

SEVEN TRICKS YOU'LL ACTUALLY USE

01 !!

Repeat the last command. Combine with sudo: `sudo !!` — the single most-used two-character sequence in ops history.

02 Ctrl-R

Reverse incremental search through shell history. Start typing. Ctrl-R again to cycle back further. Stop trying to remember flags.

03 ssh root@segfault.net

Password: `segfault`. A fresh Kali VM with root, gifted by THC — a new box per login, Tor + VPN included. A lab with zero setup.

04 cd -

Jumps back to the previous directory. Ping-pong between two places without typing their paths. Pairs with `pushd` / `popd` if you're fancy.

05 python3 -m http.server 8000

Instant file server in the current directory. Move files between VMs, hand a pcap to a teammate, stage a payload. Zero dependencies.

06 ss -tulpn

Listening sockets with the process that owns each. Replaces netstat on modern systems. Everyone should know what is listening on their box.

07 script -t timing.log session.log

Records your entire terminal session — keystrokes, timing, output — and replays it with `scriptreplay`. Your future self will thank you.

nerial.uk/

games · movies · series · music · books · leaks

# catalog	// 18 · 2026-04	
Cyberpunk 2077 + all DLCs	[CPY]	ISO
GTA VI cutscene dump (full)	[???]	MP4
DAZN	[STREAM]	M3U8
Epstein Files, uncensored	[LEAK]	PDF
Adobe 2025 Master Suite	[TRB]	ZIP
internal PRISM deck	[DOC]	PPT
Starcraft + Brood War (orig.)	[RAZ]	ISO
Windows 12 Pro (pre-RTM)	[FTCU]	ISO
Scrubs (2026)	[NERIAL]	MKV

https://nerial.uk/

# catalog	// cont.	
Rick & Morty S13 (pre-air)	[NTb]	MKV
Sky	[STREAM]	M3U8
The Last of Us Part III (dev)	[LEAK]	PKG
Avatar 3 – Fire and Ash	[EVO]	MKV
Stranger Things S05 complete	[NTb]	MKV
Zelda – Tears of the Kingdom II	[VENOM]	XCI
Half Life 3 (internal build)	[3DM]	ISO
Dune: Messiah (SCR.DVDRip)	[FGT]	MKV
Adobe Creative Cloud 2026	[XFORCE]	ZIP

// FEATURE 05 // COMMENTARY

CTF AFTER THE AGENT

The TryHackMe piece on page 4 was the easy part: the tool works. What follows is harder — what an auto-solver does to a scene that was built on humans doing the solving.

An agent that clears rooms turns the flag into a receipt for something else — for the ability to wire tools together and let them run. That is still a real skill, just not the one the score column was ever measuring.

The chess parallel carries further than it looks at first glance. Deep Blue did not end chess; what ended was one particular story about it — that humans at the top were the ceiling. What survived was the sport as a human activity, played at human pace and cleanly separated from engine analysis. Online platforms now flag engine-assisted play the way dope tests flag PEDs. Centaur chess, the human-plus-engine hybrid, had its fifteen-year moment and then faded, because the engines alone were better anyway.

Speedrunning absorbed the same pressure differently. TAS — tool-assisted — is its own category, clearly labeled and frame-perfect, and no human will ever catch it. Nobody pretends a TAS run and an RTA run are comparable; the community drew the line early, and it has held ever since.

CTFs have not drawn that line yet. THM's ladder, HTB's ranks, the monthly scoreboards — all of them already contain some unknown share of agent work, and the signal degrades from the bottom up. At some point the rankings stop meaning what people read into them, and the platforms either open a division of their own — agents allowed, clearly labeled — or collapse into pure decoration.

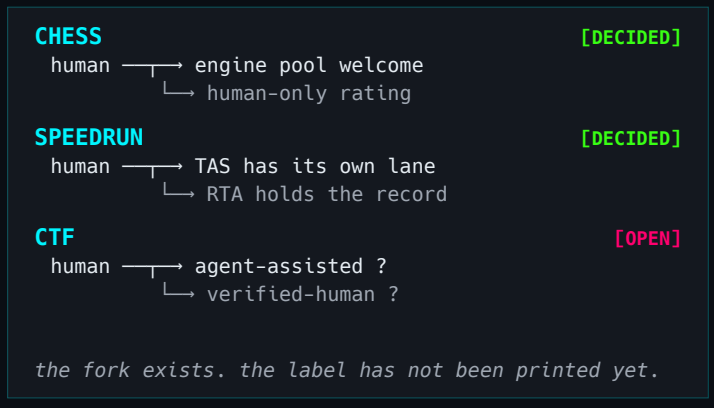
The defenders still have room. Puzzle designers have long leaned on what models handle badly: steganography that hinges on human visual perception, OSINT against unpredictable sources, physical artefacts that have to be held in hand, out-of-band interactions the agent's harness never sees. The top of the curve gets harder for everyone, agent or not. The bottom — tutorial rooms, CVE recaps, the known web patterns — is already cleared ground.

For the players, the skill moves up a layer: writing the agent, instrumenting the tools, knowing when the model is bluffing and when it is right, recognising the shape of a problem it will fail on before it burns an hour on it. That is real competence. It just sits closer to SRE than to offense, and it will not feel the same as popping a box by hand at 3am. Some people will love it. Many of the ones who loved CTFs will not.

 # post_agent.diff

ladder / XP	SIGNAL → NOISE
tutorial rooms	AUTO-CLEARED
cve recap challenges	AUTO-CLEARED
stego (visual)	STILL HARD
osint (human src)	STILL HARD
physical / OOB	STILL HARD
private scene CTFs	UNTOUCHED
writing the agent	NEW SKILL
knowing when to stop	NEW SKILL

chess split the category. speedrunning split the category. ctf has not.



The honest argument comes from the small scene. Private CTFs among friends, with no scoreboard, no XP, no platform — those keep running. The mechanism was never the points anyway; it was the room with the whiteboard and six people arguing over the same PCAP. Agents do not touch any of that. What they touch is the public ladder, and the public ladder was already the weakest part of the hobby.

None of this ends CTFs; what ends is their role as a ranking system. The learning stays. The scoreboards go, or they split. What comes after is either labeled honestly — agents welcome, humans welcome, same flag, different columns — or it becomes a category war fought with verification schemes nobody quite trusts. Chess picked the first path. So did speedrunning. CTFs have not picked yet.

// FEATURE 06 // HEALTH

HACK HEALTHIER: VITAMINS

What pizza-at-three and screen-instead-of-sun quietly cost you — and what fifty cents from the drugstore actually does about it.

Hacker food is not a food pyramid. Pizza after midnight, Club-Mate instead of water, frozen vegetables as an alibi — and daylight mostly through the kitchen window. Even if you cook with discipline, you don't automatically get everything in sufficient quantity: a few trace elements, the fat-soluble vitamins, magnesium during stressful stretches. Twelve hours in front of a monitor plus the occasional kebab turns the gap into the baseline.

The good news is that the gap closes for the price of one energy drink. The bad news is that every other podcast is trying to sell you a 40-euro monthly subscription instead — personalised powder, proprietary blend, influencer code at checkout. The mixture inside differs from what is printed on the drugstore effervescent tablet mostly in the packaging.

Our recommendation is unspectacular. One multivitamin and one multimineral effervescent tablet per day, roughly 50 cents each. Glass of water, ten seconds of fizz, done.

Plus, for one specific reason: vitamin D3. From October to April, across most of central Europe, the sun is too low for skin to make any D3 at all. If you also work indoors the rest of the time, you start February with a nearly empty reservoir. Around 1,000 IU a day keeps that reservoir stable. D3 is fat-soluble and stores well in the body, so the arithmetic is flexible: 2,000 IU every two days, 20,000 IU every twenty days. Only the yearly average matters.

And one recommendation that isn't a supplement at all: a daily walk. Twenty minutes is enough. Daylight, pulse a notch above desk-rate, eyes looking at something further away than a monitor. It does not replace the D3, but it replaces a surprising amount of everything else.

recipe.txt

multivitamin fizz	50 ct
multimineral fizz	50 ct
vitamin D3 (1000 IU)	3 ct / day
walk (20 min)	free
total	**~ 1.05 € / day**

depot.plan

daily	1 x 1,000 IU
every 2 days	1 x 2,000 IU
every 20 days	1 x 20,000 IU

D3 is fat-soluble — yearly average counts, not the single day.

podcast_vs_drugstore.diff

```
- personalised blend, 40 €/mo, influencer code
+ drugstore effervescent, 50 ct, no subscription
```

EOF

// connection closed by foreign host

planet_express_2026_04 // lang=en