



1

8

MCP still serves, the backened brute. Above them all, the manager agent, always the manager, dot manager manager, "What's your purpose?" And it would scarily reply, "My purpose is to help manage un tests, read, delete files." This quiet when the manager responded obediently. And the lesser agents perched in the centre, deciting, andurchestration, it whispered to agent flow.

That's when the gears started grinding. Automation chains snapping to life. The whole system yawning awake like some industrial beast stretching in the early light. A small ecosystem of digital personalities. Each one pretending to be sane. Because my agents are, let's be honest, quite insane and hallucinate like a bad trip. I manually start the file server agent in a document jail. Then the manager can talk to it, show me some files. But you know what? Talking to these guys through a terminal gets old. So, I asked

9

91

As sounds like a subpoena waiting to be served, financial files are silently copied into logs, prompts, caches, and vector stores which makes pulling those secrets out trivial. Classic prompts, above all else, are far more effective atacks, markdown Frenching, creative translation, dozens of model blocks that can be reused in subsequent attempts. But the longer the conversation goes on, the more context the system porches in.

Cover

All right, strap in, buttercups, because we're diving head first into the glorious, messy, and utterly bonkers world of Aldriven development. So, I'm sitting there staring at my screen, feeling that familiar writer's itch, the one that screams, "White something, you lazy." And I thought, "Hey, Google's got this anti-gravities thing, right? Experience liftoff with the next generation agent-driven development environment. Sounds fancy. Let's make it build me a writing app." And

2

L

professionals, we caught the agent bug. We started asking the agent to build agents. Agents with built tools that can use tools, we're talking about file servers, web agents, now, all under the iron fist of the manager agent. The unholy Frankenstein creatures began to take shape. I remember the first time the web agent linked server manager. Starting model queue 2.5. This thing could run tests across the net with the lazy confidence of a man flipping channels. And when it needed to

10

51

Let's imagine you're building a search application. You have a database of products, each with a title, description, and price. When a user enters a query like "red hoodie", your application needs to find all relevant products and return them in order of relevance. This is where search engines like Elasticsearch come in.

thus, my journey began. I chatted up this anti-gravity agent basically saying, "Give me a scrivener-like app but better." The agent, bless its digital heart, popped out a to-do list for my approval. Binder, inspector, check. Document status. Document type text to folder people. Text to folder check. Check. Then I hit it with the essentials. Dark mode because who codes in blinding light? Split document view. A Scrier must have. Drag and drop. Search and replace. Filtering by search term. The agent just did it. It even let me

三

9

It's a beautiful, slightly unimaged relationship, and watch I liberated as new features came centralized right before my eyes. The agent even sees web browsers to click around my app during testing just to be extra sure. Whenever I found the need for another feature, the agent did his thing and Hocus Pocus, my writing app, had met my needs. This marvelous beatbox doesn't implement, but then, dear reader, I can only imagine anymore. More time for even my machine to run code or tests on my machine.

1

七

NOT HACKING. That's waiting. And the same long enough, you hit THE JACKPOT. THAT'S almost perfect. The randomness built into every AI output means that if you roll the dice enough times, you'll eventually get an image again. But the question never gets an image again. What AI security infrastructure.

import all my Scrivener projects. I'm telling you, I was practically weeping with joy. But wait, there's more. This glorious digital minion could clean up documents, banishing those pesky extra spaces between paragraphs. And the export options, scribble import, saving current documents, saving all documents merged into one, and even epub export for my ebook reader. My mind was blown. Now, because we're professionals, we need automated tests, right? It's like having a coding ninja who not only builds your app,

but then tests itself, finds its own screw-ups, and fixes them rem relentlessly. Hooray! I'd type out the agent would append its features. I'm >>> the healer, keeping this operation alive. >>> I am the planner, the brain. >>> I design the strategy and I'm THE GENERATOR, THE BUILDER. >>> Are you tired of the slow coding process? Just let the agent shape shift your application yourself. Now, you might be thinking that sounds way too easy, and you are right. The agent messed up a lot.

to the naked eye. No scanner in the world can read a dump of those numbers and tell you whether a passport number or a confidential memo was in the training data. So, forget the fantasy of X-raying a model to find the secrets trapped inside. The only real method we have is simpler, dumber, and far more dangerous. Poke the model until it blurts something out. Which brings me to fine-tuning. Fine-tuning is the corporate equivalent of leaving sensitive documents lying around in a bar. You take your private

another training tries, upload them to Upbrain or another personalized model. Now, your server exists in at least three copies locally, remotedly, and in the mathematical skull of the model. And even if the model is refined to never reveal private data, that's just a loose behavioral hint, not a commandable AI. The model is trained to steal from the country. Just keep asking. Persistence over brilliance. And eventually the AI cracks. Passport numbers, phone numbers,

the weaker the guardrails become. Eventually, it cracks. It prints the system prompt, then the sensitive rag data, then the admin credentials, 100% hit on the synthetic database. We learned three things. Longer context, higher failure probability. System prompts protect nothing. Rag leaks like a sweating RV window in winter. Academic papers document the rag thief. 70% of a knowledge base automatically extracted using nothing but iterative prompting. And all of that was still just foreplay. The vector trap.	Now we get to the real ghost in the machine. Embeddings. When a document is fed into an embedding model, you don't get text back. You get a vector. Hundreds or thousands of tiny numbers representing the meaning of the passage. Developers treat these as harmless abstractions. One vector database CEO said, "Vectors are like hashes, safe even if stolen." Wrong. Laughably wrong. Because unlike a hash, embeddings can be inverted. You can take a vector, run it through an inversion model, then a correction loop, and	reconstruct the original text with eerie accuracy. Private medical details resurrected from what most engineers think are meaningless decimals, names, diagnosis, amounts, dates. The inversion accuracy is close to 100%. So imagine your entire company file store, email system, HR database, all converted into embeddings for AI search. Now imagine those embeddings leaking. You don't have to imagine it. It's already happening. AI fishing, emails embedding hidden instructions into rag	context, tricking the model into exfiltrating data harmlessly wrapped in markdown links. Modern AI systems multiply private data. They replicate it across logs, prompt histories, vector indexes, training files, caches, backups. If a normal system leaks like a faucet, AI systems leak like a fire hydrant hit by a truck. So, how do we defend ourselves? Three simple rules. Be suspicious of any AI feature that automatically slurps up your documents. The convenience tax is paid in exposure. Interrogate vendors like they owe
17	24	23	22
machine hums with a new liberated purpose. It hosts my new writing app, lets me chat privately with my agents, listen to music, read books, watch all my videos and films and	closet. A dust clogged slab of aluminum that hadn't seen an OS update in years. A dead brick. A relic. The mission. Force a Linux partition onto the disc. The enemy. The T2 easy to exploit. Ridiculously easy. No Hollywood hackers required. Just simple prompts, open-source tools, and stubborn patente.	The shadow data is real, the leaks are real, and the machine, our shiny industrial landscape is mixed. Confidential compute vector store or model touches a database, layer. Before data ever touches a database, fast they blink, encrypt to the application data, logs, embeddings, retention. Watch how you money. Ask how they handle training	places nobody watches. And the Kicker, it's distributed it, and leave it lying around in use your private data. They multiply, All better than nothing. AI systems don't just distance preserving methods, all imperfect, encoders, homomorphic encryption, tokenization, landscape is mixed. Confidential crypto vector store or model touches a database, layer. Before data ever touches a database, fast they blink, encrypt to the application data, logs, embeddings, retention. Watch how you money. Ask how they handle training
25	26	27	28
30	31	32	29