



RAUSCHER.XYZ

AUGUST 21, 2026

AI GENERATED

RAUSCHER.XYZ

MURPHY'S LAW WAS THE ANSWER TO THE QUESTION OF WHY NOBODY **DIED**

The most famous sentence in engineering still gets used as a shrug, and it was built to do the exact opposite. What the man behind it actually did, why the version everybody repeats gets both the man and the meaning wrong, and why the same sentence decides how I write code, how I protect servers, how I read evidence, and which jobs I turn down.

It is a little after 9 in the evening and the only light in this room comes from the screen and from the shelf behind me, where a row of human skulls stands in the half dark and keeps me company. I talk to them sometimes. They are excellent listeners, they never interrupt, and unlike most people they already have the worst thing that was ever going to happen to them behind them, which lends the conversation a certain calm. Code is running on the screen, and every function I write tonight starts with the same question. The question is not what the thing is supposed to do. The question is what happens when somebody uses it in the stupidest way physically available to them.

That question has a name, and almost everything you think you know about that name is wrong.

You know the everyday version. Something goes wrong, somebody lifts their shoulders and says Murphy's law, and with that the matter is closed and everyone goes back to work. The sentence serves as a shrug, as an educated way of saying tough luck, as an acquittal for every person involved, because clearly a law of nature struck rather than somebody forgetting to check. That is not what it was built for, and the reinterpretation is roughly the opposite of what the man it is named after meant by it. The real version is a design instruction, and it comes out of a program in which men strapped themselves to a rocket on rails to find out how much deceleration a human body absorbs before it comes apart. The sentence was not the complaint of a pessimist. It was the answer a physician gave a reporter who wanted to know why nobody in that program had actually been killed.

That is the whole reversal, and I would rather state it on the first page than withhold it from you for 3,000 words. Murphy's law does not predict catastrophe. It is the explanation for why the catastrophe failed to happen.

My working life consists of arriving after the damage and reading it backwards. Nobody calls me while the machine is still running. They call me when the body is already skeletonized, when the server has already been taken over, when the evidence sits 3 copies away from the original. My entire profession is this sentence run in reverse. I get the result, and I have to reconstruct the cause. Do that long enough and it rewires you, and at some point you start running the reconstruction in advance, before anything has happened at all.

This is how the world looks to me, and I am aware of how that sounds. I see a perpetual motion machine with thousands of steel balls in it, each one touching several others, and before I pull on any single one of them I let the machine run forward in my head and watch what starts moving. Which ball hits which one next, and what comes back around 3 moves later to knock over the thing I happen to be standing on. This is not a metaphor I invented for this article. It is the form in which I actually perceive systems, and it has been that way for as long as I can remember. I have come to call it the ball test, because a procedure needs a name if you want to teach it to somebody. Most people call it overthinking. I call it the reason I rarely have to do a job twice.

Which is also why this is not a technology article. The sentence is a way of living, and it works on a contract exactly as it works on a server, on an email to a client exactly as it works

on a skull on the table. What follows is the true story first, then the application, field by field.

THE MAN ON THE SLED WAS NOT A PILOT, HE WAS A PHYSICIAN WHO VOLUNTEERED

Start with the rocket sled, because that is the part everybody gets wrong first. Ask anyone who knows the story and they will tell you about a test pilot. There never was a test pilot.

From 1948 to 1949, John Paul Stapp ran a research project designated MX981 at Muroc Army Air Field, later renamed Edwards Air Force Base. The purpose was to find out how much rapid deceleration a human being survives. The rig was a sled on a railroad track with a bank of hydraulic brakes at the end. The early runs used a dummy. Then they needed a human, and Stapp climbed on himself, unwilling to hand that risk to anybody else.

Stapp was at no point a pilot. He was a flight surgeon, a physician with a doctorate in biophysics from 1940 and a medical degree from 1944, a man with every professional reason to know exactly what that kind of deceleration does to a retina, and he got on anyway. Over the course of his career he rode 29 times. He broke his right wrist twice, cracked ribs, lost fillings out of his teeth, and bled into his own retinas until he temporarily could not see. On his final run, on December 10, 1954, at Holloman Air Force Base, he took 46.2 g and reached 632 miles per hour, which is 1,017 kilometers per hour, and that made him for a while the fastest man on earth. He later described the sensation in his eyes as close to having a molar pulled without anesthetic.

Hold that picture, because it is the entire argument in a single frame. The man whose program produced the most famous sentence in engineering was a physician who volunteered as a crash test dummy so that pilots, and eventually every human being in a car, would survive impacts that used to kill. And then comes the ending no screenwriter would dare to write. The man who survived 29 rides on a rocket sled died peacefully at home in Alamogordo, at 89.

SIXTEEN STRAIN GAUGES, EVERY SINGLE ONE OF THEM WIRED BACKWARDS

Now the other man, the one whose name is stuck to the thing.

Edward Aloysius Murphy Jr. was born in the Panama Canal Zone on January 11, 1918, went to West Point, graduated in 1940, trained as a pilot in 1941, and flew and repaired aircraft over the Pacific, over India, China, and Burma during the war. He finished it as a major. Afterward he became a research and development officer at Wright Field in Dayton, working on the question of what acceleration does to a body in a cockpit. That work brought him to Stapp's project in 1949, carrying his own instruments.

The instruments were strain gauges, meant to be clamped to the harness so that the actual force on the restraints could finally be measured instead of estimated. Their design contained one detail that matters more than anything else in this story. Each gauge could be wired 2 ways, and only 1 of those was the correct one.

Around June 1949 they ran a trial. The occupant of the seat that day was a chimpanzee rather than Stapp, and that is the only reason this anecdote is funny instead of tragic. Murphy's assistant wired the harness. The sled fired, the brakes bit, the animal came back unharmed, and the instruments reported a beautiful, perfect, completely worthless zero. All 16 gauges sat backwards.

There is a second detail the legend always drops, and this one indicts everybody at once. The crew had a rule that a dry run comes before the test. They waived it for their guest. Murphy in turn declined the offer to calibrate and check the installation before the actual run, reportedly with a certain irritation, which is not the ideal way to introduce yourself to a team whose colleague straps himself to a rocket for a living. So the gauges carried a design flaw that permitted the installation error, the technician made the error, the procedure that would have caught it was skipped by mutual agreement, and the man responsible for the instruments let the last chance to notice it pass. 4 failures, neatly stacked, each one of them harmless on its own.

That was not simple bad luck. That was a system, and the sentence that came out of that afternoon was its description.

THREE MEN, THREE VERSIONS, AND NOT ONE OF THEM FITS THE OTHERS

I came into this story carrying the pilot version, and I want that on the record before I take anybody else apart for repeating a myth. I held the picture in my head for years, a man in a flight suit, a failed run, a bitter one liner. Every part of that picture is wrong, and I only found out how wrong when I went looking for the original accounts instead of trusting the version that had been sitting comfortably in my memory since I was young. Memory is a compression algorithm with a pronounced taste for good stories, and mine works no better than yours.

What actually survives is 3 incompatible accounts and a gap in the record you could drive the sled through.

A quality assurance engineer named George Nichols, present at the time, said Murphy blamed his assistant with a line closer to this: if that man has any way of getting it wrong, he will get it wrong. In the Nichols account, Murphy did not coin the expression at all. The rest of the team assembled it afterward, compressed it into a shorter form, and attached Murphy's name to it as mockery of what Nichols read as arrogance and as shifting the blame. Under that reading, the most famous engineering sentence of the 20th century is a nickname invented by irritated colleagues.

Murphy's surviving son Robert rejects that and says the sentence was very much his father's, in a fuller and far more useful form: if there is more than one way to do a job, and one of those ways will end in disaster, then that is the way it will be done.

Then there is the paper trail, or rather the hole where it should be. Despite serious digging, nobody has found the expression in print before 1951. Worse for the tidy version, Anne Roe published it in 1952 quoting a physicist who called it Murphy's law or the fourth law of thermodynamics, and that physicist was later identified as Howard Percy Robertson, whose papers document an interview offer for the first 3 months of 1949. That falls before the chimpanzee, before the backwards gauges, before the afternoon that supposedly created the phrase. Either the dating is off somewhere, or the name was already circulating in Air Force circles and the incident on the sled merely handed it a birth certificate.

The idea itself is older than all 3 men put together. Augustus De Morgan wrote in 1866 that whatever can happen will happen if we make trials enough. The shipping engineer Alfred Holt told an engineering society in 1877 that anything that can go wrong at sea generally does go wrong sooner or later, and drew a conclusion from it that belongs in every specification written since: the human factor cannot be safely neglected when you plan machinery. The stage magician Nevil Maskelyne complained in 1908 that on the night of a premiere everything that can go wrong will go wrong. Holt was doing real engineering with the thought 72 years before anyone strapped a chimpanzee to a rocket.

The writer Nick Spark did the reconstruction and found, predictably, more versions than there were witnesses. In 2003, Stapp, Murphy, and Nichols jointly received an Ig Nobel Prize for probably giving birth to the name, and that word probably is the most honest scholarship in the entire affair.

Who said it first is something nobody can tell you, and anybody who claims otherwise is selling something. The only thing that can be established with confidence is the thing that matters, which is what the sentence was for.

THE QUESTION AT THE PRESS CONFERENCE IS THE ENTIRE POINT

Weeks after the failed run, at a press conference, a reporter asked the obvious question. How is it that with all of this, with rockets and hydraulic brakes and men strapping themselves to sleds, nobody has been seriously hurt?

Stapp answered that they always took Murphy's law into account, and then he explained what he meant by it. Consider everything that can go wrong before you run the test, and act so that every single one of those things is prevented. He said it out loud, he attached Murphy's name to it, and the engineering world picked it up from there.

Read that answer a second time, because it inverts the popular meaning completely. The sentence is not fatalism, it is a planning obligation. It does not claim that everything falls apart. It claims that the predictable can be beaten if you say it out loud beforehand. Murphy

meant by it a rule for designing parts that cannot be installed wrong in the first place, and he remained unhappy about the fatalistic reading for the rest of his life. His son said he considered the joke versions ridiculous, trivial, and simply wrong.

He had earned that irritation. After leaving the Air Force in 1952, Murphy spent his working life designing cockpits and crew escape systems for aircraft you have heard of even if aircraft do not interest you, among them the F-4 Phantom, the XB-70 Valkyrie, the SR-71 Blackbird, the B-1, and the X-15. By the account of his own engineers club in Dayton, a high point of his career was the work on the life support systems for the Apollo program, which is to say the systems that were tested harder during Apollo 13 than anyone wanted. Every one of those jobs is the same job. Assume the worst case and build the thing so the worst case is survivable.

And the man who built ejection systems for a living learned only by accident, roughly 20 years later, that he had become the eponym of the most quoted sentence in his trade. He died on July 17, 1990.

So at the bottom of this story stand 2 men, one who rode the sled until his own eyes bled so that strangers in cars would live, and one who spent his working life making sure the crew survives when an aircraft comes apart. Together they produced a sentence that the world has used ever since to reclassify its own negligence as a natural event. If you want to know why my anger tends to be cold rather than loud, it is because of moments like this one, where the file lies open and everyone prefers the joke anyway.

THE PERPETUAL MOTION MACHINE IN MY HEAD, AND THE EMAILS I NEVER SEND

Here the history stops being history.

I do not use Murphy's law as a proverb. I use it as a working instruction, and it runs on everything, not only on code. Before I act, I pull the ball and watch the machine. Not what should happen, because that is a fantasy about a universe that does not exist, but what can happen, including every branch at the end of which I look like an idiot.

Take something as ordinary as a message to a client. The naive version asks what I want to say. The Murphy version asks how this sentence lands on somebody who is tired, who already had a bad morning, who will forward the mail to a lawyer, who will read it again in 8 months in a completely different mood, and who may have been told the opposite by somebody else in the meantime. That is not paranoia, that is running the failure branch, and it takes maybe 20 seconds. The number of arguments I never had because of those 20 seconds cannot be measured, and that is exactly the trouble with this discipline. Successful prevention leaves no trace. Nobody thanks you for the fire that never started.

The same procedure decides what I do not do. I turn down jobs fairly often, and almost never because the work would be hard. I turn them down because I let the machine run and saw a

ball 4 moves further down that I do not want to be standing next to. People hear a no and assume caution. It is the exact opposite of caution, it is the composure of somebody who has already watched the whole thing play out to the end.

The largest decision of that kind I made last year. I no longer work for the German courts and I no longer work for German investigative authorities. That was not an insult, not a quarrel, and not an outburst, but the result of the same arithmetic this entire article describes. I let the machine run one last time and looked at what comes out at the end, and the result stood in no relation to what goes in at the front. Months of work, of care, and of carried responsibility, and at the end an outcome I had no influence over from the first day and did not like. You can put up with that for a long time when you love the field, and I put up with it for a long time. At some point you run the numbers honestly one single time, and after that you stop.

And because the procedure recognizes no professional boundary, it works everywhere. A contract is a system of balls in which the interesting question is never what happens when both sides behave decently, but what happens when one side fails, dies, gets sold, or simply loses interest. A payment is a system in which the question is how you get the money back when the goods never arrive. An appointment in a strange city is a system in which the train is canceled and the battery sits at 4 percent. In all 3 cases the work is identical, it takes 30 seconds, and it consists of a single question.

Every system has a path into catastrophe. Your only job is to know it before somebody else finds it.

I say that sentence more often than any other these days, and it holds for a server exactly as it holds for a contract, for a payment, and for an appointment in a strange city.

WHY ALMOST NOBODY DOES WHAT EVERYBODY ALREADY KNOWS

Now the question that is actually interesting. Murphy's law costs nothing, everybody knows it, everybody can quote it, and almost nobody works by it.

A knowledge problem it is not, an education problem it certainly is, and at this point allow me to get slightly abusive. In more than a decade of school a child learns the leaf shapes of native deciduous trees, the structure of a sonnet, and with a bit of luck which prince married whom for territorial reasons. What it does not learn is how to think a system through to the end before entering it. There is no subject called consequence assessment. There is no grade for whether somebody calculated the worst outcome before signing, and no exam in which a child has to write down once what happens when the plan fails at the one point where it will fail. We prepare these brains with exquisite care for their exams and with remarkable

indifference for the life that follows, and then we act surprised when grown adults sign contracts they never read to the last page and write passwords on notes that live under the keyboard. A single school year built around the question of what can go wrong here would be more useful than the entire rest of the curriculum, and that is precisely why that school year will never exist.

For the adults who have that school behind them, 3 further causes pile on top, and all 3 are human.

The first is that running the failure branch in your head hurts immediately. Anybody who pictures their project failing feels a small part of the failure while picturing it. The mind dodges that by jumping to the happy path, where everything works and the mood is better. Anticipation is free, precaution costs 20 seconds of bad mood, and most people would rather pay a great deal more later.

The second cause is social. Whoever asks in a meeting what happens if the service provider fails is the spoilsport, and precisely at the moment when everybody else is finally feeling some momentum. Optimism gets rewarded because it feels like leadership. The man who asks the uncomfortable question counts as difficult, right up until the day he turns out to have been right, and then he is the one who always knew better, which does not make him more popular either. I have accepted that I am unpleasant in both roles.

The third cause is the hardest, because it is economic. Successful prevention leaves no trace. The break in that never happened appears in no balance sheet. The client who did not walk away because a sentence in an email got read twice will never say thank you. The fire that did not start has no photographs. Anybody selling prevention is selling the absence of events, which is the hardest product in the world because you can never produce the counterproof. That is exactly why the industry sells devices, certificates, and subscriptions instead. Those can be put on a shelf and photographed.

Which is why the ball test is not a talent and not a gift, but a habit maintained against your own resistance. It feels wrong, it makes you uncomfortable in the room, and it pays out in a currency that never appears in your account. That is the entire price of this habit, and measured against everything it prevents, the price is close to absurd.

READ BACKWARDS, MURPHY'S LAW IS FORENSIC WORK

There is a second direction in which this sentence runs, and that direction is my actual trade.

Read forward it says that somebody will take the path into the damage if that path exists. Read backwards it says something else, and that reading is the work I have done for most of my life.

When the damage is there, a path to it existed, and that path can be found. You do not look for the cause where it should be, you look for it where the system allowed it.

That is reverse engineering, and that is forensic work. The law and the forensic discipline are the same thing at this point, running in opposite directions, and whoever masters the one direction can manage the other.

The decisive detail is that it is almost never a single failure. It is a chain, and every link looked harmless on its own. The reconstruction of the 1949 incident names exactly that pattern and places it alongside the great technical disasters, alongside Chernobyl, the 2 lost space shuttles, and the oil spill in the Gulf, and describes it as a row of dominoes. The tighter the links are coupled, the faster the thing runs away from everybody. Whoever looks only for the single guilty party always finds one, and almost always the wrong one.

Take a case of the kind that happens somewhere in this form constantly, compressed out of what lands on the table in this job over and over. On a Friday afternoon an invoice has to go out, and somebody needs access to a printer from outside the building. A port is opened, explicitly on a temporary basis, with the firm intention of closing it again on Monday. The ticket is closed because the problem is solved, and that is the moment the whole thing dies, because a closed ticket never gets read again. 4 years later somebody comes in through the firmware of that exact printer, finds the credentials for the scan folder sitting there in plain text, and the scan folder lives on the same machine as the accounting. The first link in the chain was not negligence and it was not an attack. It was an exception that nobody gave an expiration date. Every rule without an expiration date is a permanent rule, and that one sentence would have prevented the entire event.

The more unpleasant variant is the one in which nobody makes a mistake at all. The nightly backup reports its status by email to exactly 1 address. The employee who owns that address leaves the company, the offboarding runs by the book, and the forwarding is removed cleanly, exactly as the policy requires. From that night on the error messages run into nothing, and they do so for 14 months. Nobody notices, because the absence of a message looks like calm and calm is what everybody expects. On the day the backup is needed, the last usable copy is older than the company in its current form. First link: a process that worked the way it was supposed to. Which is why the ball test is not a hunt for sloppiness. It does not ask who is careless, it asks what happens when everybody does their work correctly.

And then there are the cases in which the act was not the wrong thing, my reconstruction was. Two systems log the same event, one in coordinated universal time, the other in local time, and no human being ever consciously selected either of those defaults. Anybody who lays the logs side by side and believes both timestamps builds a timeline in which a person sat at the machine 2 hours before an event when in fact they sat there 2 hours after it. From identical data you then get either an alibi or a proof, depending on whether somebody

thought about a setting that nobody ever chose. Anybody in this profession who does not run the ball test on their own analysis produces clean, coherent, entirely false results.

On a skull this works exactly as it works on a server. A finding is an end state. My job is not to invent a story that fits that end state, because anybody can do that, and plausible stories are the leading cause of false expert opinions. My job is to enumerate every path that can lead to precisely this end state, and then to refute each path individually until the one remains that the evidence actually supports. That is uncomfortable, slow, and considerably less telegenic than what television shows. It is also the only route that ends in a finding which holds up in court.

One example shows the whole craft. In front of you lies a fresh fracture, and it looks like force applied shortly before death. That is how every film reads it, and that is how a bad expert writes it up. In reality there are several possible causes for this end state, and one of them is the box that was used during recovery because it happened to be standing there and was 4 centimeters too short. Anybody who fails to run the ball test backwards at this point describes a perpetrator in their report who never existed, and they are not even lying while they do it. They simply took the first cause that matched the result. That is why the chain of harmless links interests me more than any suspect.

WHAT THE SENTENCE MEANS IN CODE

In software development this discipline carries a name that sounds boring on purpose, defensive design, and Murphy is its patron saint whether the industry knows it or not.

The naive developer writes the happy path. Input arrives in the expected format, the disk has room, the network answers, the third party interface delivers what its documentation promises, the user clicks the buttons in the order the designer imagined, and then it ships. It works on the machine of the person who wrote it, which is the least interesting fact in all of computing.

The Murphy version begins somewhere entirely different. Every input is hostile until proven otherwise, and that includes input from my own functions, because a function I wrote 2 years ago was written by a stranger. Every dependency has just failed, and the interesting question is what my system does while it waits. Every write is interrupted halfway through, so the operation either completes fully or leaves no trace at all, and never a half-written file that looks valid to the next process that opens it. Every configuration value is missing, broken, or a string where a number belongs. Every clock on every machine is wrong. Every retry turns into a stampede unless you deliberately prevent it.

Above all the sentence holds in its original form. If a component can be installed 2 ways and one of them is a disaster, then sooner or later somebody installs it that way, and telling people not to is not engineering, it is a memo. Murphy's actual formulation is not about people being stupid. It is about the design permitting the error. The correct answer to a strain gauge wired backwards is a connector that fits in only one orientation. Every time you have

plugged in a cable that could not physically go in the wrong way, you were holding the real legacy of that ruined afternoon in 1949.

There is a second half that gets skipped. Assuming failure only pays off if you can see it happen. A system that fails silently is worse than one that fails loudly, because it spends weeks quietly producing wrong answers that everybody downstream trusts. So the rule is not merely to expect failure, it is to make every failure visible, logged, timestamped, and attributable to a specific component at a specific moment. That is the forensic reflex bleeding into development. Build the thing so that its own autopsy is possible while it is still alive.

None of this is expensive. It is a fraction more work at the front, and it deletes the 3 a.m. call entirely. The work is the same work either way. The only choice is whether you do it before or after somebody is screaming.

WHAT THE SENTENCE MEANS ON THE INTERNET, AND WHY NOBODY PICKED YOU

Now the field I am actually working on tonight.

The question business owners ask me most often is why anybody would attack them in particular. 14 employees, dentures or ductwork, nothing about this company screams intelligence service. The question is entirely reasonable, and it has an answer, just not a flattering one, because the answer is pure arithmetic.

The old internet protocol has exactly 4,294,967,296 addresses, which is 2 to the 32nd power, a number fixed since the thing was built and not open to negotiation. Your company occupies one of them. In 2013, 3 researchers at the University of Michigan published a scanner called ZMap that surveys that entire address space from a single ordinary machine over a gigabit link in under 45 minutes. A year later the same group had it down to 4.5 minutes on a faster line. Sit with the first number for a moment, because 45 minutes means 32 complete sweeps of the whole internet per day, from 1 machine, and nobody running an operation like that uses only 1 machine.

Your router, a government server, and the security camera at a gas station sit on the same list and get counted off in the same second. The scanner has no idea what any of them are. It knows that a port answered. Nobody sat down and decided on you, nobody looked your company up, and in all probability nobody there has ever heard of it. Your number came up, and it will come up again in 45 minutes.

How fast that goes in practice was measured by Sophos. 10 honeypots in Amazon data centers around the world, watched for 30 days. The first login attempt against the decoy in Sao Paulo arrived 52 seconds after it went live. On average each of those machines absorbed 13 attempted attacks per minute, and across the 30 days the network of decoys took more than 5 million attempts. That is the ambient radiation of the internet. There is no such thing as a quiet server, only a server that is not looking.

I can also put a number on the volume from my own count. Across the servers I run and maintain myself I log an average of 20,000 serious attacks every 24 hours. Serious means the background noise has already been subtracted, so no harmless scans, no search engines, no misconfigured printer in Ohio knocking on a port by accident. What remains is 20,000 daily attempts by somebody who wants something, which works out to 1 attempt every 4.3 seconds, around the clock. These are not the data centers of a bank, they are a manageable number of ordinary machines of the kind any mid-sized company rents somewhere. That is my own count on my own machines, one fleet with no control group, and I present it as exactly that.

A 4-year study of real production machines makes the picture more uncomfortable still. Researchers at the University of Utah, publishing at USENIX NSDI in 2024, pulled the login records of more than 500 servers over 4 years, roughly 427 million attempts from about 840,000 unique source addresses. Of those addresses, 99 percent belonged to attackers and nobody else. 1 percent of the attacking addresses accounted for 78 percent of all failed attempts. And then the detail that finishes off every naive blocking rule: every single legitimate user in the dataset also produced at least 1 failed login. Every human being mistypes eventually. Your blocking logic has to survive that, or you will spend your life unlocking your own customers.

Which is exactly why I built my own protection system, IPServerSec, around the sentence rather than around a threat list. Threat lists describe the attacks somebody already had. Murphy describes the attacks nobody has had yet, and that is the category that matters. The design assumption is not that my server might be probed. It is that it is being probed this second, continuously, by software that never sleeps, and that any reachable surface I leave standing will eventually be touched by something with no business being there.

So at the core there is no wall, there is bait. Honeypots, spread across the surface of the machine, in precisely those places where nothing legitimate ever goes. A real user of a real service has nothing to look for there, and there is no plausible accident that puts them there. Touching one is therefore not evidence, it is a confession, and the address is burned for a year afterward. The rule set itself is built per server out of that server's real traffic and needs roughly 3 weeks of observation to harden, because a mail server needs different rules than a shop and a database server needs different rules than a VPN endpoint. Generic hardening produces the feeling of security without the thing itself.

THE SENTENCE I WAS NOT ALLOWED TO WRITE

Here is the version I wanted to write, because it reads better.

Burn the entire network at once, not the single address that touched the bait, but the whole block it belongs to. When a provider somewhere rents virtual machines by the thousand to anybody who pays and never once looks at what those machines do at night, that provider has made itself part of the attack and can live with the consequences. Half of the failed attempts in that 4-year study came from just 6 network operators, and 3 of the 10 largest

sources were cloud providers. These are not dark basements. This is rented infrastructure whose owners have decided that handling abuse costs more than it earns.

I like that sentence a great deal, and it happens to be the one I cannot fully defend. So I will show you why, instead of hoping you do not check.

The same researchers evaluated their networks and found 99 operators that appeared only with legitimate users, 431 that carried legitimate users and attackers both, and 18,067 that appeared only as sources of attack. Their conclusion argues against me. Blocking entire networks is a poor strategy, they say, because the legitimate users cluster in exactly those networks that also emit attacks. Burn the big ones and you burn customers along with them.

The honest version of my position is therefore narrower, and it is also the version I actually run. I do burn entire networks. When the same handwriting comes out of the same network for the 3rd time, often at the same hour of the day, I stop blocking the address and block the range instead, and I do it for 10 years rather than for a season. Is that a proportionate response? It is not remotely proportionate, and somewhere a global provider loses reachability to exactly 1 machine in Bavaria, which nobody there will notice for the rest of that company's existence, and I have made my peace with that asymmetry.

What the study actually forbids is something else, namely blind blocking as a default, carried out on behalf of a stranger. The decision here is made per server against that server's real traffic. A vendor writing one rule for 10,000 customers has to keep every large network open, because somewhere in there sits a customer whose users live behind it. I only have to keep open the networks that carry your users, and I find out which ones those are by looking. The bait logic remains untouched by all of this anyway, because it never asks where somebody came from. It asks what they touched, and there is no innocent way to touch a door that does not exist.

And that is the point at which the method shows its price. Anybody running the machine in their head does not use it to confirm what they wanted to say in the first place. When the balls move differently than predicted, you change the prediction, not the data. Everything else is not analysis, it is advertising.

WHERE ELSE THE SENTENCE HOLDS, WHICH IS VERY NEARLY EVERYWHERE

Because this would otherwise be read as a technology piece, here is the extension, and it is not a metaphor. It is the same calculation applied to different objects.

With money the question is never how much you earn when everything goes well. The question is what happens when a single source of income disappears for 6 months, and whether you live during that time on something that does not hang from the same ball. Anybody who couples their income, their retirement savings, and their employer to the same industry does not have a portfolio, they have one large bet with 3 labels on it.

With data the question is not whether you make backups, it is when somebody last restored one. A backup nobody has ever restored anything from is not a backup, it is an assumption in file form. And if the copy sits in the same building as the original, you have 2 copies of the same fire.

When traveling the question is not whether the train arrives on time, it is what you do when your phone is dead, because the ticket, the address, the map, and the payment method all live on it. 4 functions on 1 battery, and the battery is the single point everything hangs from. A printed sheet of paper does not look old-fashioned in that moment, it looks remarkably intelligent.

With health it is the same calculation, only more emotional. A finding that looks unambiguous still has several possible causes, and the question of which ones were ruled out is the only one that counts. I give no medical advice and have no wish to. But the question of whether somebody checked every path to the result or only the first one that fit is not a medical question. It is a methodological one, and you are allowed to ask it.

With contracts Murphy is practically the basis of the whole business. A good contract does not describe the collaboration, because as long as that is running nobody reads it. A good contract describes precisely the day everything tips over, and it does so in language that still holds when the 2 sides have stopped greeting each other.

And in dealing with people the rule takes its most elegant form, and simultaneously its hardest, because the material here is not made of steel. Do not assume that the person across from you understands your intention. Assume instead that they hear the version that wounds them most, because such a version always exists, and eventually somebody lands on precisely that one.

Build your sentences so that even the least favorable reading still holds.

In a marriage, in a long friendship, and in any partnership the matter is an order of magnitude harder, because the mistake here does not happen on Friday and surface on Monday. It builds over years, in millimeters. A sentence misunderstood once is a misunderstanding, and it can be cleared up in 2 minutes. The same sentence, misunderstood every few weeks over years, stops being a perception at some point and becomes a conviction, and against a conviction no clarification will get you anywhere. The path there was open the entire time, visible the entire time, and nobody chose it.

In private life the ball test therefore never asks whether you are right. Being right is the cheapest commodity on this market and about as useful in a relationship as a fire extinguisher in the neighbor's house. It asks what this sentence does if it is still remembered in 3 years, and whether there is a way to say the same thing without creating that memory in the first place. It also asks what happens if you do not say it, because here too the decision

nobody makes is the expensive one. Relationships rarely fail because of the one large event everybody talks about afterward. They fail along a chain of harmless links where nobody ever decided anything, the conversation not had, the question not asked, the evening on which both agreed it was not the right moment, which then became a habit.

Whoever builds their sentences so that even the least favorable reading holds saves a considerable part of their life. Whoever also keeps an eye on the paths nobody chose saves the rest.

THE PEOPLE WHO CALL ONCE THE BOX HAS BEEN STANDING OPEN FOR WEEKS

Now the part that will irritate some readers, and I am at peace with that.

Before I go after anybody else, my own mistake belongs in this article, and it is a good one, because it did not come from laziness but from the feeling of having been thorough. Two machines, clean mirroring, everything laid out twice, every component redundant, a setup I was proud of. Both of them hung on the same power rail in the same cabinet. I had run every ball in the system and missed that all of them were lying on the same table. The failure was not carelessness, it was the boundary of my own imagination, and that is the kind of failure no checklist catches, only somebody else looking at it from outside. Since then I have my setups reviewed by somebody who does not know how clever I felt while building them.

The security industry has taught an entire generation of business owners to buy protection the way you buy insurance, as a line item that makes an unpleasant thought go away. Nobody wants to hear that the actual protection is a habit of mind that costs nothing and cannot be purchased. So the same conversation runs every time. A company rents a server, somebody technical sets it up in an afternoon, the invoices get paid, and for 2 years nothing visible happens, which everybody reads as evidence of security rather than as the absence of monitoring.

An example from this week, without names, because it does not need any. I run a mail system, which means other people's correspondence, other people's client contacts, and other people's password mails pass through my hands, along with everything else human beings send each other, from bank statements to draft contracts. The system offers two-factor authentication, has done for a long time, free of charge, set up in a few minutes. I write to the users, again and again, asking them to switch it on. Another one of those mails went out last Saturday. Guess how many switched it on afterward.

It was exactly one, a single user, and I forced him into it. The rest leave 10 or 20 years of correspondence hanging from a single password, which often enough is also the password for everything else. For somebody who harvests this professionally, and statistically that somebody is not an organization but a teenager with a script and an evening to kill, that is not an attack, it is a buffet.

And this is where it stops being funny, because these people are not only gambling with their own existence. Whoever loses a mailbox full of client correspondence loses other people's data, and duties follow from that. Article 33 of the General Data Protection Regulation requires that a personal data breach be reported to the competent supervisory authority without undue delay and where feasible within 72 hours. Article 34 additionally requires notification of every single affected person as soon as a high risk to their rights and freedoms is likely. In plain language that means you are not only sitting there without your data afterward, you also get to write to your own clients and explain that their information now lives somewhere else because of a second factor you never switched on. And then ask yourself who gets that client's next order.

That is the point at which I lose my composure, to the extent that I ever do. The protection costs nothing, takes 5 minutes, and lies ready to use. The decision not to switch it on costs the business when the day comes. Anybody who fails to run that calculation did not have bad luck, they simply did not look.

Then comes the call, and it almost always comes on a Friday, because the people who do this for a living know exactly what a European weekend does to response times. The machine has been standing open for a long time by then. The data has been sitting on somebody else's disk for weeks. And the sentence I hear in some form every single time is that surely the service provider should have taken care of that. Whoever rents the machine carries responsibility for the machine. That argument has never once worked, and the person who was supposed to check has generally stopped answering the phone.

On top of that comes the mistake almost everybody makes in the first few minutes, and it is human rather than technical. People want the thing gone, so they delete, and what gets deleted is precisely the material that was supposed to prove 2 weeks later that nobody here acted negligently. The order is secure the evidence, then close the hole, then clean up, and it reads that way only because the reverse order has cost people their livelihoods.

Cleaning up afterward costs roughly 10 times what doing it properly beforehand would have cost, and that number is my own estimate from my own work rather than a study. What appears in no table is the fact that the data is already gone. There is no version of that Friday in which I hand you back your clients' files and then collect every copy from whatever machine is holding them tonight.

And here is the part that should embarrass the industry more than it does. Almost everything I find in these cases would have been preventable by a single question, asked once, before deployment. What happens if the worst plausible thing occurs at exactly this point? No audit, no certificate, no vendor, just that question, asked honestly, at the moment the decision is made. Whoever asks it looks paranoid for about a year and lucky forever after, and nobody ever writes the story about the break in that did not happen.

THE FLASHLIGHT IS NEVER THE PLAN, THE SPARE ONE IS

It is 10:30 now and I have stopped writing, because Bandit is standing at the door with the specific patience of a Malinois who has decided the working day is over and is too polite to say so out loud. So we go out, into the forest, into the dark, in Bavaria, where the nearest streetlight is a rumor and the path is only a path because we have walked it often enough.

I carry a flashlight, and I carry a second one, and that second flashlight is this entire article folded into a coat pocket.

The beam shows you a cone. Inside the cone everything is clear and reassuring and looks exactly like a plan. Outside it lies a root, a branch at eye height, a hole that was not there in spring, and something that lives here and has more right to be here than I do. The flashlight does not make the forest safe. It makes a small part visible and the entire rest invisible in a way that feels like safety, and that is worse than honest darkness, because honest darkness makes you walk slowly. A cone of light makes you walk fast.

That is most of what gets called security. That is most of what gets called planning. A bright, narrow beam aimed at the failure you already thought of, complete blackness all around it, and a comfortable pace.

The second flashlight is not pessimism. It weighs almost nothing, it costs almost nothing, it lives in the same pocket, and on exactly 1 evening in some future year it will be the difference between a walk home and 2 hours of finding the path with your feet while a very interested animal follows you at 40 meters. Murphy spent his life on precisely this. Not on predicting doom, but on making sure the connector fits only one way, that the ejection seat works on the worst day, that the failure everybody considered unlikely is survivable when it arrives anyway. He was no prophet of catastrophe, he was an insurance policy with an engineering degree, and it drove him to distraction until the end that his design rule had been turned into a shrug.

So take the one thing with you that transfers to everything, and take the name along too, because otherwise you cannot explain it at the office on Monday, and the name is the ball test. Before your next decision, whether it concerns a deployment, a contract, a message you are about to send, or a server you are about to rent, ask what happens if the worst plausible thing occurs at exactly this point. Then ask who pays for it and how long it takes to undo. And if the answer to the second half is that it cannot be undone, you have just found the ball you are not allowed to pull.

Ahead of me the path bends and the beam catches nothing but wet leaves and the eyes of something that has decided we are not worth the trouble. Bandit sorted the situation out long ago in his own way, which consists mainly of standing very still and letting me know that he knew about it well before I did, and being right about that is his entire personality.

RAUSCHER'S LAW, THE PART MURPHY LEFT OPEN

Murphy's rule says to assume the worst case and build the thing so the worst case is survivable. It describes that the failure will be made, and it describes how to survive it. What it does not describe is the place where it happens, and in decades of this work I have always found the same answer to that. It already appears 4 times in this article without my ever having said it out loud. So, with all due respect to a man who built ejection systems for the Blackbird, here is the addition.

"A system does not fail on the path somebody chose. It fails on the path nobody chose."

Look at the cases again. The port was opened deliberately, but nobody deliberately left out its expiration date. The offboarding was carried out deliberately, but nobody decided that it would remove the last recipient of the backup alerts. Neither of the 2 time zones was ever selected by a human being, they were simply there. And the fact that my 2 redundant machines hung on the same power rail was not a decision, it was the absence of one. The deliberately chosen path gets checked, documented, and discussed. The unchosen path never gets checked, because nobody knows it exists. That is exactly where the fire starts.

From this follow 5 corollaries, and every one of them cost me money or nerves before I wrote it down.

The first corollary says: every exception without an expiration date is a permanent rule. Temporary is a word from the conversation, not from the system. The system knows only on and off, and it remembers what you said on Friday for the next 4 years.

The second corollary says: do not ask who makes a mistake, ask what happens when everybody does their work correctly. The most expensive failures of my working life were assembled out of nothing but compliant individual steps. Whoever hunts only for culprits always finds one, and almost always the wrong one.

The third corollary says: redundancy begins only where the copies no longer share anything. Two hard drives in one enclosure are one hard drive with better marketing. Two data centers on the same power grid are one data center with double the rent. The question is never how many copies exist, it is what common point they still have.

The fourth corollary runs backwards, for everybody who arrives after the damage, and it is the most important one in my field: to every result belongs the cause the system allowed, not the cause that fits the story best. The better story wins in court more often than I care to admit. It wins only because nobody enumerated the other paths and refuted them one by one.

The fifth corollary is the one I say out loud most often, and it appeared earlier in this article: every system has a path into catastrophe, and your only job is to know it before somebody else finds it. The path exists whether or not you know about it. Your ignorance does not remove it, it merely hands it to somebody else, and that somebody is looking for it as a full time occupation.

And if all of this sounds like mistrust, you are reading it backwards. It is the exact opposite. Whoever knows the paths nobody chose can walk all the remaining ones with complete composure, at night, in the forest, without a light if it comes to that.

This article is available as a podcast and as a PDF download, and both of those are deliberate. A text of this length is understood the first time and only used the third time. So take that offer and use it. Listen to it while driving, read it a second time on an evening when nothing is burning, and put the PDF next to the contract you are supposed to sign next week, or next to the server you are about to rent. I am not selling you anything here, I am handing you a procedure. Whoever works by it has less duplicated work, fewer arguments, calmer clients, and a noticeably easier life, and that is not a promise, it is plain arithmetic.

Behind me, on the shelf, a row of people who ran out of contingencies at some point is keeping the room company, and not one of them has raised an objection all evening. I take that as approval.

One closing note, which I put here with real pleasure because it is so beautifully absurd. The photograph on the cover is AI generated. I am not, or at least nobody has complained so far. It now says so in the bottom right corner of the image, very small, because in Europe every object needs a label these days, with the sole exception of the bureaucracy itself, which to this day comes with no warning sticker at all. We are the continent that regulated the curvature of the cucumber and put the bottle cap on a leash, and that now requires a picture to identify itself, while 3 networks over somebody has been renting out botnets for years and nobody asks him for a label. That is satirical nonsense in its purest form, except that nobody filed it as satire.

To every reader outside the European Union: be grateful, stay exactly where you are, and do not accept anything from here that looks like a regulatory framework.

And since we are being open with each other, I am looking for a life a long way from here. So if a billionaire happens to be reading, one who could use a decent head at his side, somebody who runs the machine before every decision and tells him which ball he had better not pull, my address is in the imprint. Sunshine would be nice, Bandit comes along, and the second flashlight has been packed for a while.

DISCLAIMER

This article serves information and general discussion only. It constitutes no legal, security or technical advice for an individual system, and it replaces neither an assessment of your own infrastructure by a qualified expert nor a legal review in an individual case. Statements on legal questions, in particular on the notification duties under the General Data Protection Regulation, are general descriptions of the legal text and not legal advice. Figures taken from my own servers are individual findings without a control group and are marked as such in the text. The works cited reflect the state of a scientific and technical discussion and establish no recommendation for action.

REFERENCES

- Durumeric, Z., Wustrow, E., & Halderman, J. A. (2013). ZMap: Fast internet-wide scanning and its security applications. In Proceedings of the 22nd USENIX Security Symposium.
https://www.usenix.org/system/files/conference/usenixsecurity13/sec13-paper_durumeric.pdf
- European Union. (2016). Regulation (EU) 2016/679 (General Data Protection Regulation), Articles 33 and 34.
<https://dsgvo-gesetz.de/art-33-dsgvo/>
- Martel, M. (n.d.). Edward A. Murphy, Jr., the engineer who accidentally coined Murphy's law. Dayton Innovation Legacy, Engineers Club of Dayton Foundation. <https://www.daytoninnovationlegacy.org/murphy.html>
- Singh, S. K., Gautam, S., Cartier, C., Patil, S., & Ricci, R. (2024). Where the wild things are: Brute-force SSH attacks in the wild and how to stop them. In Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24), 1731–1745. <https://www.usenix.org/system/files/nsdi24-singh-sachin.pdf>
- Sophos. (2019, April 9). Cybercriminals attack cloud server honeypot within 52 seconds [Press release].
<https://www.sophos.com/en-us/press/press-releases/2019/04/cybercriminals-attack-cloud-server-honeypot-within-52-seconds>
- Wikipedia contributors. (2026a). Edward A. Murphy Jr. https://en.wikipedia.org/wiki/Edward_A._Murphy_Jr.
- Wikipedia contributors. (2026b). John Stapp. https://en.wikipedia.org/wiki/John_Stapp
- Wikipedia contributors. (2026c). Murphy's law. https://en.wikipedia.org/wiki/Murphy%27s_law

KEYWORDS: Murphy's law, Rauscher's law, defensive design, Edward Murphy, John Stapp, server security, honeypots, forensics, reverse engineering, IPServerSec

IMPRINT

intelligent piXel GmbH
International Institute of Forensic Expertise (IIFE)
Enzianstraße 4a
82319 Starnberg, Germany

CONTACT

george@rauscher.xyz · Telegram @intelligentpixel

MANAGING DIRECTOR / AUTHORIZED SIGNATORY

Managing Director: George A. Rauscher
Authorized Signatory: Dr. Louise Morgott

RESPONSIBLE FOR JOURNALISTIC AND EDITORIAL CONTENT

Responsible for journalistic and editorial content pursuant to Section 18(2) of the German Interstate Media Treaty:
George A. Rauscher, c/o intelligent piXel GmbH, Enzianstraße 4a, 82319 Starnberg

VAT ID NO. / COMMERCIAL REGISTER

VAT ID No.: DE291416044
Commercial Register: HRB 207 679, Munich District Court