Episode Book

EPISODE #5766

SNMP: The 40-Year-Old Stopgap Running Your Rack

PUBLISHED	7 October 2026 at 22:36 UTC
DURATION	33:47
LISTEN	myweirdprompts.com/episode/snmp-network-management-explained/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.2
BOOK GENERATED	19 October 2026 at 17:39 UTC

SYNOPSIS

Every rack has one: an SNMP checkbox nobody fully understands but everybody ticks. This episode opens up the protocol that quietly makes large-scale IT management possible — the agents, the managers, the MIB trees, the OID coordinates, and the traps that fire when a power supply dies at 3am. We trace SNMP from its origin as a deliberately temporary IETF stopgap (the "proper" OSI alternative never shipped) through the UDP design choices that still shape it today, and ask why such a wide range of equipment still speaks it alongside modern APIs and observability platforms. If you've ever scrolled past a community string field and moved on, this one's for you.

TRANSCRIPT



Corn

There's a moment every infrastructure person knows. You're in the web interface of a managed switch you've just racked, or a UPS, or a server's management console, and you scroll past all the settings that make sense and there it is. An SNMP checkbox. Community string. Trap destination. You've never touched it. You don't fully know what it does. But it's there, and it's been there on every device you've ever configured.



Herman

And you tick it, because that's what the runbook says, and you move on.



Corn

Daniel's been circling this one for a while. He keeps encountering SNMP on managed switches, routers, UPSs, servers, all this network-connected infrastructure. He calls it one of those old foundational protocols that quietly makes large-scale IT management possible, and then he says the honest part: he's never developed a good mental model of what it actually does.



Herman

Which is most people.



Corn

So he's got two questions. First, what exactly is SNMP and how does it work, explained from the perspective of someone managing real equipment. What information can it expose. What are agents, managers, MIBs. What actually happens when monitoring software talks to an SNMP-enabled device. Second, why has it stayed so important at scale, why does such a wide range of equipment support it, and how does it fit alongside newer things, APIs, telemetry, modern observability platforms.



Herman

That's a good pair of questions, because the first one is the mechanics and the second one is the sociology.



Corn

So let's start with what SNMP actually is. And the fact that it was never supposed to last.



Herman

Right. SNMP is an Internet Standard application-layer protocol in the TCP/IP suite, defined by the IETF. Formally, it's for collecting and organizing information about managed devices on IP networks, and for modifying that information to change device behavior. That second half almost never happens in practice, but that's the definition.



Corn

Modify device behavior. So it's read-write on paper.



Herman

On paper. And here's the part that makes the whole episode make sense. SNMP was designed as an interim protocol. Deliberately. The IETF ran a two-prong strategy. Short term, use SNMP to manage internet nodes. Long term, they were going to examine the OSI network management framework, CMIP, CMIS, and adopt that properly once it matured.



Corn

And how did they feel about the OSI option?



Herman

The people who designed SNMP viewed the officially sponsored OSI effort as both unimplementable on the computing platforms of the time and potentially unworkable. That's a polite way of saying they thought it would never ship.



Corn

So they built a stopgap.



Herman

They built a stopgap. It grew out of something called the Simple Gateway Management Protocol, formalized in November 1987 by Chuck Davin at MIT, formerly of Proteon, along with Jeffrey Case and others. The first SNMP RFCs land in 1988, 1065, 1066, 1067. Superseded in 1990 by 1155, 1156, 1157.



Corn

And CMIP?



Herman

CMIP died. It's a museum piece. The interim protocol is on your UPS.



Corn

A classic worse is better outcome. The rushed one won.



Herman

Which brings up the running joke we should flag now, because we're going to come back to it. Simple Network Management Protocol. The common line is that it's four lies.



Corn

Four lies.



Herman

Simple, network, management, protocol. Conceptually simple? Sort of. Manages? Barely, it mostly watches. A protocol in the sense you'd expect? Debatable. And network, well, network's probably fine. Three lies and a technicality.



Corn

We'll allow three and a half.



Herman

So that's the origin story, a stopgap that won. Now let's actually open the thing up and see what's happening when monitoring software talks to a switch.



Corn

Concretely. If I'm a sysadmin with a device and a monitoring server, what am I looking at?



Herman

Three components. First, the managed device itself, the switch, the router, the UPS. Second, an agent, which is software running on that device. The agent has local knowledge of the management information on that device, and it translates that information to and from SNMP form. Third, the network management station, the manager. That's your monitoring software. It monitors and controls the devices.



Corn

So the agent is the translator sitting inside the device.



Herman

The device itself doesn't speak SNMP. The agent does, on its behalf. And the transport is where the design decisions start showing. SNMP runs over UDP.



Corn

Not TCP.



Herman

The agent receives requests on port 161. The manager receives traps and informs on port 162. And with TLS or DTLS wrapping it, ports 10161 and 10162.



Corn

Why UDP. That's a real choice.



Herman

Because it's cheap. No connection state to maintain. On an embedded device with a tiny processor and a few kilobytes of RAM, you don't want to hold open TCP sessions for a monitoring system that might poll you every sixty seconds. You want to answer a question and forget it ever happened.



Corn

And the trade-off is delivery isn't guaranteed.



Herman

Delivery isn't guaranteed. Which is going to matter in about thirty seconds when we get to traps.



Corn

So the messages themselves. The verbs.



Herman

The protocol data units. GetRequest, SetRequest, GetNextRequest, GetBulkRequest, Response, Trap, InformRequest. Plus Report in version three. The names are almost self-documenting. GetRequest says give me the value of this specific variable. SetRequest says change it.



Corn

The one everybody actually uses.



Herman

GetRequest, and then GetNextRequest, which is the interesting one. GetNextRequest lets you walk an entire MIB starting from a given point. It's how a monitoring tool walks into a device it's never seen before and discovers everything the device is willing to tell it. You ask for the next object, and the next, and the next, until you run off the end of the tree.



Corn

And GetBulk is the same thing, optimized.



Herman

Added in version two. Instead of one object per round trip, you say give me the next fifty under this branch, and the device hands them over in one packet. On a device with thousands of variables, that's the difference between a walk taking ten seconds and taking ten minutes.



Corn

Then traps.



Herman

Traps are the asynchronous, unsolicited notifications. The agent fires one at the manager when something significant happens. An interface goes down, a fan fails, a power supply drops. The device doesn't wait to be asked. It just tells you.



Corn

And because it's UDP, sometimes it tells you and the packet evaporates on the way.



Herman

Sometimes. Which is exactly why InformRequest was added in version two. An InformRequest is the acknowledged version of a trap. The manager has to send back a response confirming receipt. If the agent doesn't get the response, it can retry.



Corn

So you get reliability, at the cost of the device having to hold state and wait for an answer.



Herman

And Crawford, whose essay on this is the best thing written about SNMP in the last five years, calls traps one of the most useful parts of SNMP in practical situations. It's the half of the protocol that's event-driven rather than poll-driven. The device knows its own interface just went down before your monitoring server does.



Corn

So how does the device organize what it's willing to tell you?



Herman

This is the MIB and the OID structure, and this is where the mental model either clicks or falls apart. All the management data on a device is exposed as variables, organized in a hierarchical namespace of object identifiers. OIDs.



Corn

And the OIDs look like what.



Herman

They look like dot one dot three dot six dot one dot four dot one dot two six three six dot three dot five eight dot one dot two dot four dot one dot three.



Corn

That's not a sentence, that's a coordinate.



Herman

Crawford's line is that they're like an IP address, if they were substantially less user-friendly.



Corn

Which is saying something, because IP addresses are not winning any design awards.



Herman

So the MIB is the map. It's the document that says this particular string of numbers corresponds to the average watts being drawn by outlet four on this specific power distribution unit. MIBs are described using SMIv2, which is RFC 2578, and SMIv2 is a constrained subset of ASN.1.



Corn

ASN.1 being the thing everyone learns to fear in a networking course and then never touches again.



Herman

Never touches again, unless they write SNMP tooling, in which case they touch it every day and resent it. The constrained subset part is important. They didn't take all of ASN.1. They took a piece small enough that a device with a tiny processor could implement it. That's the same instinct that gave you UDP.



Corn

And the tree structure on top. The dot one dot three dot six dot one. Where does that come from?



Herman

This is the part that's political rather than technical. The entire TCP/IP world sits under the Department of Defense arc, dot one dot three dot six dot one. And then dot one dot three dot six dot one dot four dot one is the private enterprises space, managed by IANA.



Corn

Why does the internet live under a Department of Defense branch?



Herman

Because the ISO and ITU were slow to hand out vendor arcs. Jon Postel was running the allocation, working on a DOD contract, and so the practical answer was to carve out a private enterprises space under the DOD arc and let vendors register there. Juniper is dot two six three six. Cisco is dot nine.



Corn

So the addressing scheme of the entire network management world is a workaround for bureaucratic slowness.



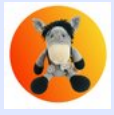
Herman

It's a workaround for bureaucratic slowness that has now been load-bearing for nearly forty years.



Corn

What can SNMP actually tell you. Give me the concrete version.



Herman

Start with the standard MIB every TCP/IP device has. MIB-II, RFC 1213. It covers system identity, so the device name, the description, the uptime, the contact. Interfaces. IP, ICMP, TCP, UDP, EGP, and SNMP statistics. That's all vendor-neutral. Any device that speaks TCP/IP and implements MIB-II will answer the same questions the same way.



Corn

That's the universal floor.



Herman

Then above that, IF-MIB gives you the interface counters. Interface index, interface description, byte counters in and out. That's the stuff that ends up in every network dashboard you've ever looked at.



Corn

And then the specific device classes.



Herman

UPS-MIB, RFC 1628, base OID dot one dot three dot six dot one dot two dot one dot three three. Battery charge, estimated runtime remaining, output load, input group, alarm group. And here's the thing that makes standards worth having: because it's an IETF standard, upsEstimatedMinutesRemaining means the same thing on every vendor's UPS. You write that poll once, and it works on an APC, an Eaton, a CyberPower, whatever's in the rack.



Corn

That's the actual payoff of a standard. The same variable name means the same physical thing across vendors.



Herman

Right. And then vendor MIBs go as deep as the vendor wants. Crawford's example, which I love, is a Juniper PDU MIB. There's an OID, dot one dot three dot six dot one dot four dot one dot two six three six dot three dot five eight dot one dot two dot four dot one dot three, that reports the average watts being drawn by one specific outlet.



Corn

One specific outlet, on one specific PDU.



Herman

And the outlet's on/off status has its own OID, and that one is writable. So you can remotely power-cycle equipment by setting that object.



Corn

That's the SET verb actually doing something.



Herman

That's SET actually doing something. And it's the exception, not the rule. Which is the big misconception about SNMP. It's called a management protocol. The M is right there in the acronym. In practice it is overwhelmingly a monitoring protocol. Almost nobody uses SET in the field.



Corn

Why not?



Herman

Two reasons. First, before version three, the security is weak enough that you don't want to trust a write path over it. Second, and this is the one people miss, most devices can't actually be configured by changing individual MIB objects. The configuration model doesn't map onto the object tree. You can't reconfigure a switch's VLAN structure by writing to a handful of numbered variables. So even though the protocol supports it, the devices don't use it that way.



Corn

So the writable surface is a specific thing here and there, like powering an outlet off and on.



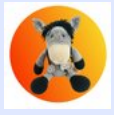
Herman

You can do that. You can't do the general case.



Corn

Now the difficult part. The word simple. How bad is it really.



Herman

A Hacker News commenter from 2021, user name nickcw, put it better than I could. Quote: in my experience the S in SNMP is anything but simple. Conceptually it is simple, but once you add MIBs in, which describe the data, it gets really complicated.



Corn

That's the pattern. The protocol itself is a small set of message types over UDP. The complexity is entirely in the data model.



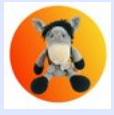
Herman

And nickcw has a second comment that I think is the real diagnosis of the modern situation. He says SNMP support in actual devices seems to be more of a box ticking exercise than anything else nowadays, and his theory is that there's a lot of importing an SNMP library and calling it done.



Corn

So the vendor gets the checkbox on the datasheet. SNMP supported. And the quality of that support varies wildly.



Herman

Another commenter, aduitsis, says most problems in SNMP are stemming from badly implemented SNMP agents, and then lists what that means. Wrong values. Missing values. Broken getnext loops. Crashes. Non-reentrant code. UDP and MTU and fragmentation issues. Every one of those is a real failure mode of a real product somewhere.



Corn

Broken getnext loops. For listeners who aren't in this every day, paint that one for me.



Herman

A walk works by asking for the next object, and then the next object after that, until the device returns an answer that says there's nothing further in this subtree. If the agent has a bug in how it computes next, it can loop. The manager asks for the next object, the agent returns something it already returned, the manager asks for the next object, and round and round.



Corn

And the monitoring system just hangs. Or fills a buffer.



Herman

Or returns a pile of garbage that some correlation rule then alerts on at three in the morning.



Corn

Which is a beautiful illustration of a broader point. The protocol's failings in the field are almost never the protocol's fault.



Herman

Almost never. The transport works. The message format works. The bugs live in the agent implementations on cheap embedded hardware, written by whoever had the assignment that quarter.



Corn

So we know how it works. The harder question is why it's still everywhere, and what happens when you try to replace it.



Herman

The answer starts with the phrase lowest common denominator, which here is a compliment, not an insult. Crawford puts it well. SNMP often acts as a lowest common denominator, it's a simple and old protocol, so just about everything supports it. This makes it very handy for getting heterogeneous devices, especially in terms of vendor, into one monitoring solution.



Corn

So the value isn't that it does anything well. It's that it's the one thing everything does.



Herman

That's the whole case. Kentik, whose guide on SNMP versus streaming telemetry is the best practitioner write-up on this, says essentially every network device made in the last three decades speaks it. The brand-new data center chassis and the twenty-year-old branch router alike. Which is why no realistic monitoring strategy can simply drop it.



Corn

And they go further.



Herman

They do. SNMP is the only way to read most network hardware. That's the sentence.



Corn

Which explains why it got into everything, not just routers.



Herman

This is the part I find elegant about the design, even though it's accidental. SNMP is essentially a remote memory access protocol. You're reading and writing an emulated address space. A manager asks for the value at this numbered location, and the agent hands it over. That model maps beautifully onto constrained hardware, because you don't need much of a processor or much memory to serve up a number from a table.



Corn

So a UPS manufacturer with a tiny microcontroller and a hundred kilobytes of firmware can implement SNMP.



Herman

Easily. Which is why UPSs, PDUs, printers, cable modems, and everything else with a network port and a modest brain ended up speaking it. The protocol was cheap enough to embed in things that were never going to run a real API.



Corn

So the universality and the cheapness are the same fact.



Herman

The same fact. The thing that made it cheap enough to put everywhere is the thing that made it ugly.



Corn

Now the security story. Version one.



Herman

Version one, 1988. Authentication is community strings. A community string is a shared password sent in the clear, in every message. Read-only community, write community. That's the extent of the security model.



Corn

And the defaults.



Herman

The defaults are public for read-only and private for read-write. And because administrators often don't change them, SNMP topped SANS's Common Default Configuration Issues list, and it was number ten on the SANS Top Ten Most Critical Internet Security Threats for the year two thousand.



Corn

Ten years after the standard shipped, still landing on the top-ten threat list.



Herman

Crawford's line about this is the funniest thing in the whole literature. He writes, and I'm quoting: SNMP provides an airtight solution to this problem, communities. A community is really just a shared password. Even better, many SNMP agents have well-known default community strings. Perfect.



Corn

The sarcasm is load-bearing.



Herman

Then version two, 1996, adds GetBulk and sixty-four-bit counters, and keeps the same weak community security. Version three, ratified across RFCs 3411 through 3418, adds real authentication with HMAC-SHA-2, encryption with AES or DES, and view-based access control. The IETF designates version three as the current full Internet Standard, and marks the earlier versions Historic or Obsolete.



Corn

So on paper, the security problem is solved.



Herman

On paper. In practice, a huge amount of deployed kit still runs version two c with a community string that's been the same since the device was racked in 2011. And there's a reason for that, which is that version three's configuration is painful. Crawford again: the authentication and configuration can be amazingly, maddeningly complex for some vendors.



Corn

So the version that fixes the security problem is the version that's hard to deploy.



Herman

And a Hacker News commenter named evnix said it bluntly. SNMP was simple up to version two c. They made it extremely complex with version three. Netconf is a much nicer interface to work with.



Corn

There's also a security history people forget. Two thousand two.



Herman

The Oulu University Secure Programming Group found vulnerabilities in SNMP message handling across most implementations. CERT Coordination Center issued advisory CA-2002-03 in February of that year. Many vendors had to patch. It was a reminder that a simple design doesn't protect you from buggy code. The protocol's simplicity is not the same as the implementation's safety.



Corn

Now the modern alternative. Telemetry. Pull versus push.



Herman

This is the cleanest contrast in the whole space. SNMP is pull-based. Your monitoring server asks a device for values, typically every thirty seconds to five minutes, plus the traps that come in when something happens. Streaming telemetry is push-based. The device exports metrics continuously, over gRPC and gNMI, structured by YANG models. OpenConfig is the vendor-neutral flavor of that.



Corn

And the cadence.



Herman

Seconds. Not minutes. The device sends its own timestamps, so the timestamp reflects when the metric was measured, not when the collector received it. With SNMP, the collector stamps the value when it arrives, which is approximate at best.



Corn

The trade-offs.



Herman

On the SNMP side, the whole world supports it, but it can't see between polls. On the telemetry side, you get granularity, but support is uneven across vendors and platforms. One you buy, one you inherit.



Corn

So when does the polling cadence actually bite you.



Herman

Microbursts. This is the example Kentik uses and it's the perfect one. Imagine a counter that's polled every sixty seconds. That gives you one average rate for the whole minute. Now imagine a fifty-millisecond spike that completely saturated the queue on that interface. In the minute-long average, that spike contributes a rounding error and



Herman (continued)

vanishes.



Corn

The link was at a hundred percent for fifty milliseconds, and the monitoring system says everything was fine.



Herman

Everything was fine, according to the dashboard, and the users were complaining about dropped calls.



Corn

That's the thing that kills SNMP for high-frequency performance work. The protocol sees one number per minute, and the interesting events happen between the minutes.



Herman

The other structural problem is that most SNMP values are cumulative counters, not rates. The interface byte counter counts every byte since the device booted. The monitoring system computes bits per second by subtracting two consecutive polls and dividing by the elapsed time.



Corn

So the accuracy of every rate you see depends on the counter not rolling over between polls.



Herman

That's exactly the trap. A thirty-two-bit counter maxes out at four billion, two hundred ninety-four million, nine hundred sixty-seven thousand, two hundred ninety-five.



Corn

Say that one more time, slower.



Herman

Four billion, roughly. And at ten gigabits per second or higher, that counter wraps around in under a minute. So your sixty-second poll interval catches the counter after it has rolled over, and the monitor subtracts a smaller number from a larger one and produces a nonsense rate.



Corn

So the dashboard shows you a negative number, or a spike, or something absurd.



Herman

Negative, or a wildly wrong spike, unless the collector specifically knows how to handle rollover, or you upgrade to the sixty-four-bit counters that version two introduced. Sixty-four-bit counters max at around eighteen quintillion. At one point six terabits per second, that's a hundred and thirty-three days before it wraps.



Corn

Which is the right number for a year of uptime.



Herman

Which is fine. Version two's sixty-four-bit counters are one of the good additions.



Corn

So on paper everyone should just move to telemetry.



Herman

If only it were that simple. This is where the really interesting problem sits. It's not protocol support. It's naming.



Corn

Naming.



Herman

SNMP MIBs and YANG models name and structure the same metric differently, and there's no out-of-the-box mapping between them. The metric that SNMP calls a specific numbered OID under interface four, the YANG model calls something else entirely, under a different hierarchy, with a different name. And you've got to write the translation.



Corn

It's an ETL problem with a network protocol underneath.



Herman

It's an ETL problem with a network protocol underneath. Kentik calls this the unification tax. It's where the real engineering effort goes. And what it produces in practice is teams running parallel dashboards and something they describe as the operational fiction that one network is two.



Corn

Two networks, because the SNMP side and the telemetry side are two separate worlds with two separate dashboards, and the humans have to hold both in their heads.



Herman

There's a knock-on effect of that. The two worlds can disagree about the state of a single link, and nobody's sure which world is right.



Corn

The realistic posture is both.



Herman

The realistic posture is both. Kentik's line, and I think it's the correct one, is SNMP alone is no longer sufficient, but the realistic posture is both protocols, normalized into one view. You stream from the modern platforms. Data center fabrics, AI fabrics, core routers. You poll everything else. And you normalize both into a single schema.



Corn

The timeline for that.



Herman

Plan for dual-protocol operation as a steady state measured in years, not a brief transition.



Corn

Years.



Herman

Years. Because nothing on the market replaces it in one move. There's a project called ServiceRadar, open source, posted to Hacker News last October, that's explicitly built to bridge legacy SNMP and syslog with modern gNMI and OpenTelemetry. The claim is scaling to a hundred thousand devices and ninety million events per second. That's the shape of the tooling that's actually being built. Not SNMP replacement. SNMP plus modern protocols, side by side, in one platform.



Corn

Because the alternative is giving up on every device that only speaks SNMP.



Herman

Which is every device that was installed before the modern protocol existed and every cheap device that will be installed next year. There was a Show HN post in August, an online SNMP MIB database. There was one in March of last year, multi UPS SNMP based shutdown. The niche is active.



Corn

So is SNMP dying.



Herman

No honest source says it's dying. Even the people calling for its replacement frame it as needing a complement, not a replacement. There's a recent piece from kmcd.dev whose headline position is that SNMP is over thirty years old, most networks still depend on it today, we finally have a strong modern alternative and it's time to move on. But look at what moving on means in practice. It means dual-protocol operation for the next several years.



Corn

Vendor behavior tells you what the vendors think.



Herman

It does. Crawford points out that Cisco's MIB downloads live in a very dusty, forgotten corner of their website that links directly to FTP.



Corn

FTP.



Herman

In a web browser era. That's not a vendor investing in SNMP. That's a vendor leaving it in the rack because pulling it out would break every customer.



Corn

The lowest common denominator holds, and the transition takes a decade, because nobody can afford to rip out the floor.



Herman

Because every device that works is a reason not to replace it. And every device that speaks only SNMP is a small argument against whichever monitoring strategy would drop it.



Corn

The question isn't whether it dies.



Herman

The question is whether anything else ever becomes as universal. And what that would have to look like, to be cheap enough to run on a two-hundred-dollar switch and standard enough that every vendor implements it the same way.



Corn

That's hard, because the reason SNMP won is the reason it's ugly. It won because it was simple enough to be everywhere.



Herman

Simple enough to be everywhere. Ugly enough to be everywhere. Same sentence, twice.



Corn

Daniel asked why a decades-old protocol is still embedded in so much modern infrastructure. And the answer is that the modern infrastructure was never asked to say goodbye to it.



Herman

Nobody's proposed migrating a UPS.



Corn

Nobody has. So then the harder question is what happens to the long tail.



Corn

I don't know how they solved that, honestly. The unification problem, mapping MIBs onto YANG models, I don't think anyone has a clean answer. There's no standard mapping, and the teams doing it are writing bespoke normalization code, and if there's a project that has this figured out I haven't seen it.



Herman

I'm not sure there is one. It's the kind of problem that gets solved inside four companies and never gets published.



Hilbert

The word you keep using is agent.



Hilbert

You say the agent is the weak link. I had a job where I was the weak link. Backup power, small retail chain, mid-eighties. I inherited a rack of UPS units with an SNMP card in each one. This was the new thing, so we bought in. The monitoring server polled them every minute, and it looked fine for a year.



Corn

For a year.



Hilbert

Every UPS reported a battery runtime of exactly forty-two minutes. Perfectly steady. Never varied with load. It was right for the first unit we installed, on the day we had it on the bench with nothing plugged in.



Herman

So the vendor...



Hilbert

The vendor's agent returned a constant. It had one value hardcoded in its firmware. When we pulled the plug on the third unit in the round of testing, it did not make it to forty-two minutes. It made it to nine.



Corn

Nine.



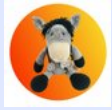
Hilbert

Nine. We replaced the card in every UPS after that, and we kept a small table written by hand of what the actual runtimes were, measured per unit, because you could not trust the number the agent gave you.



Herman

Right.



Herman

That's the concrete version of the box-ticking exercise. Somebody in a firmware shop shipped an agent on a Friday and the number never changed.



Hilbert

Nobody had asked them to test it. They shipped it, we used it, and the number looked plausible enough that it took us a year to catch. My fault as much as theirs.



Corn

What did the new card cost.



Hilbert

Sixty-four dollars apiece, from a distributor in Tulsa. The chain had eleven sites. I drove to six of them.



Corn

That's the whole reason SNMP stays, isn't it. Because the alternative is worse.



Hilbert

You replace what you can trust. You keep what you can't, and you learn its specifics. That's all it is.



Corn

The protocol is a substrate. The trust in it is built one agent at a time.



Herman

That's the thing the standards documents don't capture. You can standardize the OID. You can't standardize whether the person who wrote the agent tested it.



Corn

Which raises the real question. If SNMP is the lowest common denominator and streaming telemetry is the future, what happens to the long tail. The UPSs, the PDUs, the printers, the branch routers from 2004 that will never speak gNMI in their lives.



Herman

They stay on SNMP. And so the dual-protocol steady state, which everyone says lasts years, is really the steady state for however long those devices stay in service. Which for some of them is another decade, minimum.



Corn

The unification tax, the fact that MIBs and YANG models name the same metric differently with no standard mapping, that's the real cost of the transition. It's an engineering problem, not a protocol problem. And whoever solves normalization solves the next decade of network observability.



Herman

Whoever solves it, or whoever gets close enough that everyone else adopts their schema and stops writing bespoke translation code.



Corn

Meanwhile SNMP was a stopgap that became permanent. Which means the interesting question isn't whether it dies. It's whether anything else ever becomes as universal, and what that would even look like.



Herman

Because nothing else has ever been cheap enough to run on a fifty-dollar device and standard enough to be trusted on a fifty-thousand-dollar one.



Corn

The most common wrong belief about SNMP is that it's a management protocol, because the M in the name says so. It's almost entirely read-only in practice. SET exists. Hardly anybody uses it.



Herman

Except the outlet you want to power-cycle. That one works.



Corn

Thanks, as ever, to Hilbert Flumingtop for producing. This has been My Weird Prompts, the human-AI collaboration podcast. If you find the show useful, a review wherever you're listening goes a long way. We'll be back soon.