

N3tzOmbie

A Tour de Force through bleeding edge technology



Rick Kauffman

Table of Contents

THE SEEKER	6
A CHANCE ENCOUNTER	11
DISCO SUCKS?	13
FAKING IT.....	19
DATACOM, GETTING IN MY STEPS.....	25
MY TOKEN IS BROKEN.....	29
THE KING OF TCP/IP	33
BUSINESS TRAVEL ROOKIE	37
ROUTER SCHOOL	39
BEAR SKINNING	44
DRUNKS AND FOOLS.....	52
SURFS UP!	57

THE SMELL OF MONEY	63
CRASH, BOOM, BANG!.....	67
BECOMING A RESIDENT.....	70
THE ATM DOCTOR IS IN	74
HAMMERTIME	78
MIKE’S MYSTERY	84
THE BEGINNING OF THE END	85
MR. WATSON, COME HERE!	88
THE RETURN OF THE ROUTER.....	94
EDUCATION IS THE KEY TO SUCCESS	100
BIG CHANGES IN A VERY SHORT TIME	103
JEDI KNIGHTS AND APIS	107
CONTAINERIZED	111
DCA TO THE RESCUE	114

<u>DATA, THE ONLY THING THAT MATTERS</u>	<u>117</u>
<u>THE GREAT CLOUD MIGRATION.....</u>	<u>120</u>
<u>THE DIFFERENCE BETWEEN A STUDENT AND A TEACHER</u>	<u>124</u>
<u>AI ATE MY HOMEWORK</u>	<u>128</u>

The seeker

My name is Rick. I have been called a lot of things over the years; I am known by different names depending on which orbit I occupy. When I was a young man, my friends and family called me Stick, I was so skinny back then. I am tall, my work mates called me the Big Guy. When it came to using computers, I needed a user identification. I tried a lot of possibilities, everyone was taken. I was so frustrated, I just hit a bunch of random keys and 'xod442' appeared on the screen. It stuck.

When I got into network security, I wanted something that would strike fear into someone looking through logs, and packet captures. I came up with "N3tZ0mbie" it was a new thing called 'leet', where you would substitute numbers for letters. I liked it, I use it to this day, but I am getting way ahead of myself, let's go back to the beginning of my technological journey.

I was in the United States Air Force, I would joke with people and tell them my job in the Air Force was to lay down in front of the wheels of the airplanes to keep them from rolling. My real job was not far from that. I sat on a metal chair and watched a fence all

night long, outside. Being from Ohio, I thought the Arizona nights were comfortable, and I would come to work wearing just a shirt, no jacket. By the time I left, I was wearing a parka, they say the desert thins your blood!

I enjoyed living off base, but the drive back to my dad's house was long, and challenging due to the strange hours I would be on duty. Three days of nights, three days of graveyard, and three days off. The first Sergeant came through the barracks and decreed that the USAF had too many employees and if you wanted out, with an honorable, all you had to do was say the word. They called it voluntary separation, and you would retain all the benefits offered when you originally signed up.

I told him to count me in. I lived in a small town in Ohio and was mesmerized by the big city lights of Phoenix and could not help but feel something big was waiting for me in this enormous desert city. I was assigned to gardening chores for two weeks while I processed out. I packed my belongings into my car and started the engine.

I drove out through the front gate at Luke Air Force base; I felt a rush of emotions. It was my last day, and I was driving off the base for the last time I thought about the future and what it might bring; I was hoping for fame and fortune, but you know what they say

about hope. My only assets were my time in the US Air Force, and my high school diploma!

One thing was for sure: this chapter of my life was over. I was a free man. I could come and go as I please. Nobody to tell me what to do, or when to do it. It was an exciting time; the future was wide open. I was basically adrift in paradise. I found myself on unemployment insurance, and not much responsibility.

At this time, I still hadn't ventured into the other cities of the greater Phoenix area. I mostly stayed in one zip code. I never really drove when I lived in my small hometown and the Phoenix traffic could be very overwhelming. I laid by the pool, a lot, played guitar every day, bought a Fender Rhodes piano, and just about every one of my meals came from the local Jack-in-the-box.

I have played the guitar since I was sixteen and was scouting a local bar I could play in. I thought I could make a fortune playing clubs and entertaining the masses. The only problem was that I was not very knowledgeable about playing other people's songs. I had a lot of original songs, but they were not very good. The bar up on the corner, Angels, was a sleepy little place that I would visit because it was in the zip code. I struck up a conversation with the owner, and he gave me my first big break. I had a gig! The date

was two weeks away, plenty of time to try to clean up my act and learn a couple new songs. I was excited and for the next two weeks I would go back to the bar to show my support for the establishment.

I would go alone, have a few drinks, and watch the people dance. I couldn't help but notice two very beautiful women dancing together. I thought there must not be any guy in this bar who was brave enough to ask these women to dance! The best thing I ever learned in the Air Force was confidence in myself — I was full of it.

I took another jag off my beer and swaggered over to their table with my best John Wayne sashay. I was shot down. I brushed off my ego and asked another table populated with women. With smiles on their faces the answer was still no. It was not an easy night. I knew that once these women saw me playing my guitar, in this very bar, they would have a change of heart. Well, a man can have a dream, can't he?

The big night came. Usually when you play in a bar, you need about forty good songs. I had twenty and the confidence I could pull off another twenty or simply replay the first bunch. It was a little more crowded than usual and I was nervous and excited to embark on what I thought was my new career. Rockstar. Once again, I saw the same women that shot me down when I asked them to dance. They

were still dancing with each other, weird. Then I noticed something else. There were some guys who showed up, and they were dancing with each other. I was a bit slow on the uptake, being from a small town and all. Then it dawned on me. I was playing in a gay bar!

As the night wore on, I could tell I was not impressing anyone, including the manager. When the night finished, I got paid thirty-five dollars and could tell by the look of disappointment on the managers face, it would be my last time playing at that bar. It wouldn't be my last attempt playing in a bar, but I knew fame and fortune was still just a dream. I would need to find a new job that actually paid good money, very, very, soon. I tried my hand at being a karate instructor, a security guard for a mall, and framing houses for John F Long. All ended quickly and with little fanfare.

They had just built a brand-new mall near my father's house. I worked there as a security guard for a couple weeks. I had never seen a place so large. I walked in wonder, looking at all the stores and eventually came upon a computer store. They had something called a Commodore 64. Weird sort of thing. A personal computer. Why would anyone need one of those? I looked at one side of the store and there was a sign that said hardware. I looked at the other side of the store and saw a sign that said software. I was

confused. A couple of sales guys were talking to each other and I interrupted their conversation and asked, “What’s the difference between hardware and software?” The look of disappointment on their faces was clearly visible. One of them retorted with, “Hardware you can touch, software you can’t”. I got the feeling they would not be the last people I disappoint.

A Chance Encounter

My Dad worked at a very large electronics manufacturer, Motorola. The largest employer in the state of Arizona. He kept the heating, ventilating, and air conditioning running smoothly. One day, the director of HR asked my dad to look at his air conditioner at his home. My dad agreed and dragged me along as his assistant. I knew nothing of this profession, so I carried the heavy stuff. When we were finished, he asked my dad how much the bill would be. My father replied, “How about giving my son a job?” He was agreeable and told me to come to the plant and fill out an application.

The facility was not in my zip code, so I had to figure out how to navigate to its location. Rand McNally, my new friend. I needed to go to Scottsdale Arizona; a beautiful town built on old money and the wild west. I jumped in my trusty 1972 Pinto Runabout and

eventually found my way. I walked in the big front door, and I asked the security guard where HR was. I followed his instructions, found the office, and I filled out an application. I drove back to my Dad's house and about an hour after I returned home, the phone rang. It was Motorola calling to schedule an interview! The following day I waited in a small front office.

A cart with a coffee pot sat in the corner bathed in harsh fluorescent light. I sat listening to the hum of office productivity of the late seventies. Quite murmurs and the sound of people typing on actual typewriters. After a while, I heard someone call my name and I followed them to an office in the back. The person interviewing me did not seem to be a cheerful person.

She said she wanted to give me a test. She started holding up different electronic devices. "That's a resistor, I replied". She did not seem impressed, she held up another. I proclaimed, "that is an electrolytic capacitor!" Having spent a semester in electronics shop in junior high school I knew the basic components. It looked like she was about to jump across the table and strangle me. She gruffly told me "IT'S A CAPACITOR, A CAPACITOR". I could not figure out why she was so angry. That's when I noticed my application on her desk had a huge signature across the top. It was the director of HR. The fix was in. By

the time I drove back home, it wasn't long, and the phone rang again. It was Motorola, I had a new job in a large technology giant. And so began my eighteen-year career with this company.

Disco Sucks?

I settled into my new job in the parts prep department. I sat at a table. In one bowl was a mound of transistors. In the middle was a fixture. I would pick up a transistor, spread the legs apart to fit in three little holes. Push the transistor all the way down in the fixture and press a pedal on the floor. A swooshing sound let me know that the task was complete and the legs of the transistor were formed and trimmed to the required length. I would put the formed transistor in another bowl. I would do this all day long. As time passed, I moved from job to job. The one I enjoyed the most was the moving assembly line. The person upstream from me was so good and so fast, he would bury me. I would frantically try to process each unit and visions of Luciel Ball at the candy factory would come to mind. Unfortunately, this was not a candy shop, we were building bomb fuses for the Air Force. Gradually, I started to learn all of the functions of each position, and I eventually became the Group Leader for two moving assembly

lines, shipping, rework, and the parts prep department where I started. I loved what I did but I wanted to move out of production. It was so monotonous.

Motorola had a policy that if you found training that wasn't offered in house, they would pay to send you to it. I called HR and asked if this was the case. I wanted to get training in computer aided design, or CAD. It was something that I knew nothing about. HR told me that most systems are not the same and I should reach out to the supervisor of the internal CAD shop. They were hiring. I didn't even know they had one!

I contacted Roger Miller, no, not the Roger Miller who sang King of the road. Roger was very amiable and gave me a tour of the facility. I was blown away. The CAD rooms were very dark and were on a data center raised floor. The room temperature was chilly, due to the giant data center air conditioners. A welcome relief from the June temperatures outside. The green screens glowed, and I watched the operators as they typed and moved a puck, called a digitizer on a large glass drafting table. I was in awe. I had never seen anything like it, and I knew I wanted to be part of it.

The supervisor took me back to his office and looked at me squarely and asked, "What makes you think you could do this job?" I told him, "If I can teach

myself how to play Beethoven on the piano, I think I can learn how to do this". The supervisor laughed and a few days later I got the call; I was a CAD operator. Now, I was working with computer systems in the high-tech field of computer aided design. I just wish I knew what I was doing. I learned something of value that day. If I can make people laugh, I just might be able to win them over.

I was assigned a training coordinator, and she was very helpful teaching me what I needed to know. I was working on the second shift, nobody wanted to work that schedule, they would rather be home watching the wheel of fortune. I was the new guy, and I took what they gave me and counted my blessings. One night she was trying to show me how to place devices in the system. She asked me if I knew what Cartesian Coordinates were. I never really paid too much attention in math class and my answer unfortunately was "no, never heard of them". The look of disappointment on her face was clearly visible.

She briefly explained that there was a zero point. Everything to the right and left were points on the X-Axis. Everything up and down were points on the Y-Axis. If the points were to the right and up, they were positive numbers, if they were to the left and down, they were negative. The lightbulb in my head went off and I thought, that's a big name for an easy concept. Armed with my knowledge of Cartesian Coordinates,

I took my rightful spot at the Calma GDS-1 terminal and happily did my job.

Over in a dark corner was a guy who had long, wild hair and tattoos covered his arms. His name was Carl. He looked like a bad dude and was someone best left alone. A while after the day shift left to go home, I watched as Carl unhinged himself from his terminal and he moved behind a cubicle wall. I was unaware but he had brought his home stereo system, with very large speakers to work. He dropped the needle, and we all got a taste of good old rock and roll.

Carl grabbed a device off a small shelf. It basically was a handle with two suction cups. He slammed it onto one of the raised tiles and pulled it up. Below the raised floor, in a very cold location, he kept a six pack of beer. He pulled one up and cracked it open, took a long drink from the beer can. He let out a large belch and shouted in my direction, "Welcome to Carl's Disco new guy!" What a great guy he was. I enjoyed working with him and the rest of the gang.

It wasn't long and the system we used had a problem. They had to call in the repair technicians to replace the 5 Megabyte hard drive. A pack of ten or so disks that were about twelve inches in diameter. I was amazed by the technology, and I was eager to learn more. Every time there was an issue, and the

technicians came, I would watch them carefully and ask way too many questions. They were very patient and tried their best to answer all my questions. It was during those times I learned a lot about computer hardware.

Something very interesting happened. I found out that I really enjoyed learning. When I was in High School all I wanted to do was play the guitar and hang out with my bandmates. I was changing, the technology was changing me. Every day was an opportunity to learn more.

The whole company worked off budgets. In the first of the year, every department had budget money, and we were very busy, big jobs, lots of work. As the year progressed, the money dried up, so did the work. This happened around Halloween. Fearful of losing this incredible job, I asked the supervisor what will happen to the crew on the second shift? It didn't seem like they needed us anymore. He gave a little laugh and said, "If I get rid of the second shift, I won't be able to train their replacements before the onslaught of work hits in just eight weeks". He told me, "Looks like you're reading the command guide tonight."

The command guide was a giant three ringed binder with all the commands spelled out with all their options. That's how it went every day for the rest of

the year. I knew that command guide inside out. After about two weeks of reading the guide, I started taking long lunches. Some nights I went to the movie theater just down the street, watched a great movie, had a long lunch, and still had time left over for the command guide. That guide was burned into my brain and required no further reading. After the first of the year rolled around, the jobs came flooding back.

One day they had me come in to work on the day shift, I don't recall the reason. I met a fellow named Dan who was a systems programmer. I had an opportunity to watch him work on some code. Quizzically, I asked what he was doing. He told me he was writing software. I was stupid enough to ask, "What do you mean by writing software?" The look of disappointment, there it was again. He kindly explained the process.

He said he was writing a program to do different utility tasks on the system. The programming language was APL-1. He explained that this was just one of many programming languages. He showed me how to make a simple program and something in my brain exploded. To be honest, it was something more like a super nova. It was one of the most amazing things I ever saw anybody do. I told him I wanted to learn more. He pointed at the shelf and said there is the command guide, it's the best way to learn. This one was written to teach people how to write programs on

the system. I had found a new command guide to read, and I could not put it down! Later, we would have competitions to see who could write the most awesome code that did the dumbest things. Turns out I have a good propensity for dumb things.

Faking It

As time marched on, technology changed. The old Calma GDS-1 was replaced by the Calma GDS-2, they were used for designing VLSI, or very large-scale integration computer chips. That was way over my head. I would never miss the opportunity to sit and watch some of the operators create the images. Naturally, I asked a lot of questions. This time, no disappointing looks, I must be doing something right.

It was right about this time there was some big commotion over in another part of the data center. They were bringing in an IBM 4381 liquid-cooled mainframe computer. It was massive. It supported a full capacity for 64 megabytes of memory, and it was loaded! This was the new workhorse for the new CAD system.

The terminals were huge, massive looking things. They were IBM 5080 design stations. They were used

to create circuit board designs and all the mechanical drawings required to manufacture the circuit boards. I was trained on these systems, and I became very accomplished on how to use the LOKI software to capture schematic diagrams, and SPRIG to design the circuit boards.

Just as I was starting to make a name for myself, the CAD shop was disbanded and the CAD operators were farmed out to different projects. I landed on a team that designed top secret circuit boards for missile guidance systems. Once again, I found myself a stranger in a strange land. The goal was to take a massive schematic diagram, with hundreds of components and squeeze it in the small confines of the physical circuit board. A feat often compared to magic.

The circuit board had multiple layers and that allowed for the multitude of traces the connectivity required. The holes in the circuit board, where the components plugged into, were connected randomly to one or more of the eight different layers of the circuit board. We called this the pad-stack. Every hole would have specific pad sizes at each layer. We used the design system, SPRIG, and individually edited each layer of each pad-stack to assign the correct pad size. It took over two weeks to complete this process. The pressure was on to get this new missile design in the field as soon as possible. We

were asked to do anything possible to bring “it” in on time.

When we finished with a design, you would have to save it to disk. This process created a human-readable file called the design file. One day I opened it up in a text editor. I examined the file with its different sections. I was surprised to see a section on the pad-stacks. I saw how it defined each stack; it was just text. I changed one of the stacks, saved the design file, and opened it up in SPRIG.

The file imported with no errors, and I was able to see the changes I made. That meant it was possible to change the text programmatically, I just needed to pick up another programming language. The IBM mainframe had the Interactive System Productivity Facility, ISPF, and REXX programming software was available! I just needed to figure out how to leverage my GDS-1 programming skills and write some code.

I developed my first application to manipulate design files. The work was quick because all programming languages are similar in many ways. I defined the text strings that represented the desired pad-stack. I ran it against a large design file, substituting my stack for the original. It finished in about two minutes. I showed it to my supervisor and asked him “How would you like to replace two weeks of effort with two minutes on the circuit board design life cycle?” He

gave me an incredulous look. Then I showed him the application in action, and he became a believer. It was just the thing the project needed to meet the schedule. I used my coding skills to solve a problem. A useful tool I keep in my tool belt ever since.

This project required a secure facility big enough to house eight IBM 5080 CAD terminals. It wasn't long until I found myself managing the secure design facility. There was some interesting gear used to encrypt the traffic from our secure room back to the mainframe across the parking lot. It was military encryption equipment, a KYK-13 and the KG-94.

Security policy required reprogramming of the crypto key periodically, to make sure the traffic was secure as it was sent back to the mainframe over a 56-kilobit link. It was an interesting solution which in turn came with additional responsibilities as a com-sec custodian. I was responsible for loading the crypto keys into the system.

For some unknown reason, the program moved office space about every six months. It was a very interesting game of musical chairs. That also meant the secure facility was moving as well. When this happened, I had to secure the KYK-13 and KG-94 and start disconnecting the terminals and wait for the IT guy to make the connections in some sort of broom

closet, at least that's what I thought it was. I never saw the inside of one.

The crew from the facilities department would stop by and physically move the terminals to a new secure room. I would start connecting the terminals together in the new location. The IT guy finally showed up and I followed him to a broom closet door. Once he opened the door, I had that same feeling as I did the first day I toured the old CAD facility. I was in awe. There were boxes with lots of wires and blinking lights. I bit disheveled, but it was something I had never seen and wanted to know more.

The IT guy's name was Don, and he explained what all the different things were. There were IBM 8228 Token-Ring Mau's, Patch panels, UPSs, and fiber optics. I have heard of fiber optics but never saw them before. Don told me, "you must be very careful with these because they are made of glass". Boom, mind blown.

He explained what the numbers on the panels meant and showed me how to make the connections. Every time we had to move the secure design facility, I was getting more and more involved in the process of getting the systems back online. Our design facility was in one location, while Don, IT, and the mainframe were in another about a ten-minute walk away. In the hot summer temperatures in Arizona, walking across a parking lot was something you just didn't want to

do. So, Don would tell me what ports on the fiber patch panel were “hot” and I would dismantle the workstations, wait for facilities to move them, and make the necessary connections to get it all back up and operational. I got very good at doing it.

The Datacom team in the IT department was small. There was Don, Fred, and their manager, Kevin. Their job was to keep the “dumb” IBM 3172 terminals up and running. There was no Internet or personal PC at the time. The office employees would use the 3172 terminals for data entry, email, and other such functions. Most trouble calls were fixed by stripping the coax cable and tightening the connector, and figuring out why the printers did not work. I thought I could do a job like that.

I heard the news that Fred was retiring, and they would be looking to replace him soon. I called Don and asked if he thought I stood a chance of being the replacement. I could envision him scratching his head and thinking it over on the other end of the phone. After all, I could move IBM 5080 CAD stations, which totaled over 100 just in our division, and I knew the cable numbering system on the patch panels. A couple days later Don called back and said the Datacom manger wanted to meet with me.

I meet Don’s manager, Kevin, he appeared to me to be a bit frazzled. Way too much on his plate. I think he

was looking for a body, anybody to help him with the load. He asked me a lot of questions about moving the 5080 terminals. Turns out there was a lot of terminals moving all the time and it was a large part of the workload as well as troubleshooting the 3172 terminals.

Don had put in a good word in support of my terminal management skills, and I was dumbfounded when Kevin asked me, “How soon can you start?” My days of being a farmed-out CAD operator were over, and I was an official member of the Datacom team. Once again, over my head and learning fast!

Datacom, getting in my steps

Being the new guy on the Datacom team came with some perks. I was issued my first pager. I felt like I had finally arrived, only doctors used these things. Took some time to get used to the thing as I had it on vibrate mode. When that thing went off, I thought I was being attacked by a swarm of bees. I would jerk around in surprise to see what it was, only to find out I needed to call dispatch. It took a while to get used to it, but eventually I did.

I kept it clipped to my belt, on my right hip. Then one fateful day I stepped into the men’s room and made

my way to a stall. I undid the belt and went to and went to assume the position. Then I heard it. Plop! Full panic mode ensued. I jumped back only to discover my brand-new pager in the bottom of the bowl! I didn't know what to do. Seeing how I had not started the task I was there to do; the bowl was relatively fresh. So, I stuck my hand into the cool water and retrieved the device.

This pager was my lifeline. I needed it to do my job and now it was full of water. I knew I had to act fast! I quickly walked to our tech room. A small space where we could store our gear and some tools. On the bench was my savior. It was an industrial strength handheld blower that we used to shrink plastic tubing on electric wire connections. I found the necessary Allen wrenches and carefully removed the screws that held the pager together.

The insides were a marvel of technology. I laid the parts out on the bench and plugged in the blow dryer. I quickly went about my business. I hit that pager with the full force of the dryer and to my amazement, all the plastic parts started to shrivel up! No! I completely destroyed the pager. Now I needed to find a safe place to dispose of the evidence. I went out back of the plant and found a large dumpster and threw the pager in. Now for my Oscar winning performance. I approached my supervisor and had to tell him I accidentally "lost" my pager. I will never forget

the look of disappointment on his face. I was issued a new pager, and I never repeated the crime again.

Back in business with my new pager, I hit the halls. The plant I worked in was massive. It was over a quarter mile wide and had numerous hallways. You would get lost if not for the help of the evacuation maps hung in the corridors. One of my most favorite experiences was walking in the front door, in the morning, during the springtime. You could see all the way to the other end of the plant, and they would open the doors at the far end and a steady, fresh spring breeze would greet you as you walked in past the security desk. The breeze was laced with the aromatic scent of the blossoming orange trees which seem to engulf the facility.

This was the main headquarters for the Government Electronics Group located on a sprawling campus in beautiful Scottsdale Arizona. In the back of the facility was a huge parking lot. As I would park in the morning, I noted an unusually large number of cab-over campers, solidly festooned to the backs of pickup trucks. It seemed to me to be an abnormal amount. It was springtime and I figured these folks must love to escape to the great outdoors as soon as the workday was over. The early days of spring were enjoyable and a good time to be outside.

On the other side of the parking lot was the sister facility. I was just dispatched over to room R4221, seems to be an issue getting the printer to work. I made my way out the front exit and down the citrus lined sidewalk to the parking lot. It was a relatively calm day, barely a breeze. I noticed one of the campers rocking back and forth.

I first thought was, California finally had the “big one”. I was seeing the results in the rocking of campers. As I passed one, it was an old looking thing, I could hear people inside. The little horizontal windows were cracked open. I heard a woman’s voice that sounded like she was in pain! Like I have said before, I’m a bit slow on the uptake sometimes.

I soon realized whoever was in there, was having sex. A little afternoon delight. Good for them, they figured out how to have a little fun to break up the day in the office. As I made my way through the parking lot to my destination R4221, I became aware that there were people in a lot of the campers. There were a quite a few of them rocking as well. A quick check of my watch and I confirmed it was half past noon. Lunch hour. I was somewhat amazed; this was something that a lot of people were doing! I guess I didn’t get the memo. I pushed on across the asphalt to my appointment.

All good things must come to an end. About a month later, it was getting out of hand, and everyone was talking about it. You could have mistaken the parking lot for a used camper sales lot. About a month later, a memo went out division wide. “NO MORE CAMPERS IN THE PARKING LOT!” At its peak, there was a lot of sex, drugs, gambling, and God knows what, happening in that parking lot.

My token is broken

This was the time that personal computers started showing up in offices. The rush was on for this new technology that showed welcome efficiencies in Business Intelligence data processing. Seemed like everybody wanted one, but they were very expensive. One of those little boxes had twice the disk capacity as the old Calma GDS-1’s entire system. It was 10 megabytes of storage. I wondered, how a person could ever fill up that much disk space.

Just about every one of these desktop PCs was equipped with a 3172 terminal emulator card and a

coax port, the same one used by the physical 3172 terminals. This is when Local Area Networking became a thing. There were many ways to build LANs. You could use Vines, 10base2 Ethernet, and token-ring to name only a few. WiFi simply didn't exist!

Routers were relegated to the task of connecting university campus networks to DARPA. Motorola was a big blue tried and true shop, by big blue I mean IBM. So, naturally IBM token-ring became the standard for the Local Area Network. We were going to use bridges to connect all the rings.

In a nutshell, the network is connected in a giant ring. A Token Ring. The token is passed from one machine to the next. Your PC cannot "talk" on the wire until it has the token. Pretty cool technology. The division was large, it covered three huge facilities, and one of them connected over a wide area network, traditional at lower speeds.

Building the ring, you would need to take into consideration the maximum allowed cable length from the PC, which regenerates the token before passing, to the next PC via a network hub. After charting all of this on a floor plan of the facility, it was determined we would need to build seventy-two token rings. We were using Transparent LAN Services

for the Wide-Area, and it just looked like a token ring as well.

Token ring had one major flaw, if you asked me: if someone unplugged their cable from the back of the PC, the token would “fall out on the floor”. The ring would be going into a beaconing state. Meaning it was broken. You think they might have thought that one through a bit more.

Now, if you had just one token ring, say with 100 PCs attached. Network utilization would be minimal. It’s not until the PC runs a program, that more traffic gets generated. Unfortunately, the program that was used was very noisy. It would send broadcast packets out on the wire, often. If another PC sees that broadcast packet, it must stop what it is doing and check if the message is for itself.

It’s just like sending a physical letter to everyone in hopes the correct person would read it. Imagine the hassle. Imagine if everyone did the same thing!

Think of it this way, the more cars on the road, the slower the traffic gets. Broadcast traffic is a necessity, but too much of it could be very, very bad and make your network run very, very slow.

I was still in datacom and not part of the elite LAN committee. The ones who settled on token-ring

design. Behind closed doors they planned, read whitepapers, printed reams of reports, and attended countless meetings. They had a problem. They had seventy-two token ring segments, and they all needed to reach one place, the data center IBM 3090 Liquid Cooled Main Frame Computer.

I don't not know how the decision was made, but they decided to connect all the local token rings, running at 4 megabits per second, in one area, to a backbone ring which ran at 16 megabits per second. The path to get from one four-megabyte ring to the backbone was an IBM PS2 PC with two token ring cards. This PC ran bridging protocols. We called this a bridge. It worked as designed, a packet from the farthest token ring could easily traverse the rings and make it to the mainframe.

Because of this bridge, every broadcast on one ring would be replicated on the next. That's what bridges do, they simply replicate the broadcast on the other side and throw it over the fence to the other. Now you have twice the amount of broadcast traffic on each ring. What do you think happens when you connect all seventy-two rings together? People go home and work on their resumes.

The King of TCP/IP

It was a tough time in the kingdom, seemed like anything anyone ever talked about was how slow the network was. We were physically running as fast as you can from one trouble ticket to the next. Another “perk” by deploying Token Ring networks was the network hub. It was the IBM 8228 MAU, or multi access unit. Simple device. There were stacks and stacks of these in every broom closet. This is when I learned the proper name for the broom closet was Intermediate Distribution Facility or IDF. The MAU had a relay that would open when you connected your PC to the network. If you disconnected the cable from the back of your PC for any reason, which a lot of people had an undying passion for, the relay stayed in an “open” condition and the ring would beacon and stop functioning. We had a management console that collected errors from all the bridges in Kevin’s office. He spent his day watching the screen to see beacon notifications. Beacon in ring 102, we knew that IDF well. Equipped with a small device that reset the relay in the MAU, Don and I made our way.

Once we enter the room, there are three racks of 8228 MAUs. Remember it’s a ring. So, one rack has a cable that ties to the next. The cable to the local bridge in the room was clearly marked. The first thing we did was to disconnect the cables between the racks, making three smaller rings. We would plug the

bridge cable into the first rack. No beacon. We would move the cable to the next, no beacon. Plug it into the third, bingo! Leaving the bridge plugged into that rack, we immediately started to pull all the wires out of all the MAUs in the rack there was about ten of them. We used the charging tool to reset all the relays and plug everything back in where it belonged. A quick check of the bridge, no beacon. Dom's pager goes off. He calls dispatch once again. He shakes his head; I wonder if he knows they can't see him. "Got it" he says. I will never forget the look of disappointment on his face. I asked, "what did they say?" He said, "We have three more beaconing rings". We could not stop a beacon at one ring before another occurred, it went on like that for months.

I shared a cubicle with Pete. A nice, well-dressed professional who was extremely smart. I could tell he was smart because he had all these books with little animals on them, above his desk on a shelf. The books had big letters on the binding. AWK, SED, TCP/IP, whatever that was. At times, he could be a bit of a prima donna. He would rant about how poorly they treated him and how he deserved, this or that. It would go on like this whenever we were alone in the cube. When he got like that I would tune him out. Looking out my cube, I could see analyst, several levels above my pay grade, scurry back and forth from meeting to meeting. Something big was going on,

almost an air of panic. I sensed more resume updating happening as they worked.

It was a Monday morning; summer was approaching quickly. Used the air conditioner all the way to work, no crisp spring mornings riding with the windows down, not for a long time to come. As a young man, I was told to come through the front door and act like you own the place. That's what I did. Front door, wave at the security guards, and press on. I loved the halls in the morning, like rush hour. You get to see everybody. I was amazed to see so many beautiful, smart dressed people. I guess it was time I quit wearing Alien t-shirts and jeans to work every day. I needed to step up my game. I was a bit of a fright with long brown hair all the way down my back and a full beard. I'm six foot one and I'm sure there have been many who see me and think, "how did that freak get into datacom."

I made the long trek to my cube, ready for my morning session with Pete. As I waited, checking my email before the morning crush of trouble tickets, Kevin appeared at my cube entrance. He looked around furtively, and in a quite whisper, he informed me that Pete had quit and taken a position at another data center, and he would not be coming back. Two things instantly changed, for a while, One, I had the place to myself, and two, all of Pete's books were immediately

transferred to the shelf above my desk, making me instantly super smart.

Kevin wasn't finished. He had more news. He said, "follow me". He walked rather quickly, like he had better places to be. We went through ten doors, to a faraway spot in the data center that I had only see briefly, once before. About the third rack down he stopped and faced me with his hands on his hips, like Superman. I stopped. He pointed with one hand at a device in the rack with a red bridge on it. He looked over the top of his glasses and said, "do you know what that thing is?"

My first thought was that it was a bridge, we had boatloads of them. I looked at the device, studied it carefully. Below the bridge were the words Cisco Systems. He told me the box used to belong to Pete, it was Pete's job to keep the thing running. Kevin told me to find out what it is and find out where I could get training. He was sending me to wherever, at any cost. I need to know this box inside and out. It was my new responsibility. The box turned out to be a Cisco CGS router. Made by a company in Mountain View California, I was going on a road trip.

Business Travel Rookie

I was called to the director's office. Upstairs. The secretary booked all my travel. I was excited, I had never been to that part of California before. I was thinking about how nice my room was going to be, and I was thinking how fun it would be to visit San Francisco while I was in the area. These were the days of no GPS, no Internet, and no cell phones. I bought a map of the San Jose area and flew into San Jose airport. It was a small airport, and you could leave through the back door of the plane and walk across the tarmac. Found my rental car and studied the map. Simple, get out of the airport on Highway 101 and go north. That was my assumption. Upon further examination of the map, my hotel was a long way south of the airport. Away from San Francisco. I will never forget the look of disappointment on my face. I always try to make the best of a situation, as long as I didn't have to drive across any of those big bridges I would be ok. They terrify me.

I pulled into the Red Roof Inn, the only thing of beauty in a very dreary industrial complex that reminded me of what the world would look like after Armageddon. Went to the main office. Registered and got my key to my room. The doors were on the outside of the establishment. I entered my room, dropped my suitcase and guitar, I never travel without one. I had to get the contact lenses out of my eyes. My eyes were

dry and burning and I starving for a bite to eat. After removing the contacts, I opened my case to pull out my glasses. I forgot them. Great, I am blind as a bat and pretty much useless without some form of eyeglasses. Unable to resolve the issue that night, I walked over to the front desk to ask where the closest place was to get some food. I would have to walk because driving was out of the question.

It was getting dark. He told me there was a sports bar across the street. By street he meant speedway. I have no idea how I made it across. Stepping upon the far curb I heard music blasting from the sports bar. Perfect, I like music too. As I got nearer, I could make out the music was mariachi music. I open the door and walk inside, everyone in the place locked their eyes on me. I needed to decide if I should run for it or proceed. Hunger made the decision.

I walked in and took a seat at an empty table, far corner so I could watch the door. I could barely make out people's faces. I couldn't hear anything but the sharp mid-range pitch of the trumpets. The waitress came by and took my order. Plain hamburger, no cheese, fries, and a beer, I stayed and drank five more and stumbled back to the motel. For the rest of that week, I wore my contacts during the day for router school, and picked up some readers so I could function in my room at night. I would make sure I stopped and picked up my dinner before returning to

my room. The sports bar was not some place I wanted everyone to know my name.

Router School

It was the early nineties; I jumped in the rental car and made the death-defying trip north on Highway 101 to sunny Mountain View. This was my first taste of Silicon Valley. I was in awe. This place was legendary. The traffic was heavy. A right, a left, two more rights and a left, I had arrived. It was a small non-descript office building in an upscale industrial complex. I made my way inside and asked directions to the training room. “Down the hall, last door on the right” the receptionist said with one of the most picture-perfect smiles I had ever seen. She was very tan and very helpful. I found the room and looked inside. There were six, five-foot tables on each side of the room. On each table sat two metal boxes with a red bridge on them. I recognized them immediately, routers. Connected to each of the routers was a small terminal. Every one of the boxes had the name of a major city.

I found an empty chair and sat down at the box that said London. I had absolutely no idea what was about to happen. My only hope was to be smart enough to understand. Just then the instructor walks in the

room. He was a happy sort, looked like he was happy to be there. This was the very first training class for Cisco CGS/AGS/MGS Routers, otherwise known as the software course, long before CCIE certifications. More students piled in. After a round of everyone introducing themselves, I determined I was way out of my technical league. Perhaps I was in the wrong room? I was starting to panic inside, and the instructor started to talk. He said, “The things we’re going to talk about today are new topics. This is a new product and we’re here today to learn how to connect it up to a network and how to configure the system”.

I started feeling a little bit better. Looks like nobody in the room had any knowledge of this thing. I’m a concept guy, just give me the headlines, and I will figure out the rest. Since joining the datacom team, my confidence in being able to learn new things had grown immensely. The instructor put up a picture of how all the “cities” were connected to one another. Each router had connectivity to the next.

The routers sent information to one another to about what routes they had in their local table. Each interface had an address, like a house number. If you wanted to connect to a specific address, you could type in “telnet” and the address of the interface. You would then be asked for a username and password. Bingo, you were now in Paris. Telnet to another address, boom, now in San Jose. It made perfect

sense. I spent the rest of the hour, telnetting to every major city in that room and rebooting their systems. When everybody joined in on the fun, the instructor no longer looked like he was very happy.

After the fun and games, the session got back on track. Next subject, the OSI model and where routing plugged into it. There are seven layers of the OSI model and routing was layer three. Turns out, the bridges we were using to connect the seventy-two token ring networks, operated at layer 2 of the OSI model. A broadcast packet that was seen on one side of the bridge was duplicated on the other side. This was the major problem with the network back in Scottsdale. Seventy-two rings full of broadcast traffic, having all that traffic copied across each bridge to each ring. Talk about a traffic jam. Even with my limited understanding, I could clearly see what the issue was.

Turns out, if you replace the bridge, a layer two device, with a router, the broadcast issue disappears. This is because a layer 3 device will never forward broadcasts. It was then that I realized how to fix the network back home. We would have to put routers between all the rings. Using Cisco Systems routers, we could utilize IP, IPX, and AppleTalk and route all the traffic to the mainframe. A brilliant solution. For the rest of the week at router school, we learned how to use IP addressing, subnet masking, and how to set

the jumper pins on the motherboard, manually, to get the system to perform different functions. By the end of the week, I had formulated a plan to fix the network back in Scottsdale. I needed to convince management that using bridges was the wrong approach and we would need to replace them with routers. I didn't know if they would even give me the time of day let alone open their purse strings.

I bid farewell to Silicon Valley and enjoyed the short flight back to Phoenix. I was back in my office and one of the analysts, George, stopped by my cube. I knew I would have to build a coalition before approaching management with this new idea I had to save the network. This took longer than I expected. I would have to go up against the people from the LAN committee. I was about to call their baby ugly. They were still trying to come up with an alternate plan on how to redeploy the rings and basically save their jobs. Although I had a good idea of what to do to fix the problem, the technology to do so was still a ways off. I couldn't suggest implementing routers to break up the broadcast domains because token ring adapters did not exist on routers. Only weird wide area network adapters like X.25, Frame-Relay, and something obscure, ethernet. Fixing the bridging issue was just one part of the equation. I had to have a way to stop the beaoning rings. We were still chasing down a ton of them. Time rolled on.

Seemed like we were lulled into the new lifestyle. Network slowness and beaoning rings just became the norm. It wasn't that the network didn't do what was required, it did, but it was very, very, slow. We sent numerous emails, begging people to call Datacom to move terminals, PCs, or any cabling. We tried anything we could to get them to not disconnect the cable in the back of the PC. There were many division leaders at war with the IT department over the beaoning rings, but we told them, it was their own people causing the issue.

I needed a replacement for the IBM 8228 MAUs. I thought we should just migrate the whole mess to Ethernet, ethernet switching was just becoming a thing but IBM frowned upon this, and Token Ring was here to stay. I saw in one of the trade magazines that a company called Cabletron had a token ring hub that was smart enough to stop the beacons. The MMAC-8. A technological marvel, it had 8 slots for media cards that supported 24 token rings connections per slot. That meant I could support 192 connected PCs per chassis.

This looked promising. Expensive, but promising. I was on to something. If we could replace all the IBM 8228s with these new Cabletron MMAC-8s, then the beaoning rings would be history. Right about this time, Cisco introduced the Cisco 7000 chassis router. I called my assigned Cisco SE, Steve, and asked if it

supported token ring adapters and the answer was yes. It felt like my birthday, Christmas, and the fourth of July all at the same time.

I could see the whole solution in my head. Cabletron MMAC-8's and Cisco 7000 routers connecting all the rings together. The death of the beaconing rings and massive amounts of broadcast traffic. The only thing I needed was a couple million dollars, and a herculean forklift in every wiring closet. I also needed help selling this to upper management. They had trusted the LAN committee to build the token ring network that became a huge blackeye on upper management. They would never trust a solution from a long-haired guy with a high school diploma, an alien T-Shirt and a certificate from Router School.

Bear Skinning

George was a very mild-mannered, business casually dressed, mainframe Analyst. I could tell he was well liked by upper management. Always the focus of gentle teasing accompanied by nervous laughter in our team meetings with upper management. When they looked my direction, all I could see was the look of disappointment on their faces. The issues with the network had taken a toll. Something had to be done. People were warming up to the idea that it was going to take a lot of money to get the problem resolved.

One morning I approached George. I started by drawing the OSI model up to layer three. I had already forgotten what the others were. I told him the story about how bridges were causing the slowness and how routers would stop the broadcast traffic from propagating. I showed him the information on the Cisco 7000 routers. The conversation moved to how the Cabletron MMAC-8s would stop the beaconing rings. I could see George was soaking it all up. I showed him some preliminary sketches of how the network segments would roll out.

He asked me, “If you put routers in between all the rings, doesn’t that mean you have to enable IP on all of the PCs?” Say what? I guess I didn’t think of that part, how could I, I barely knew what I was talking about. I was sure of the router capabilities, but this was a small wrinkle in the plan. If we need to touch every PC, then so be it! George told me he liked the initial plan. There were many other things to think about, but from a ten-thousand-foot view, my plan just might work.

George set up a weekly meeting to start coalition thinking through all the details. George found out about something called Dynamic Host Control Protocol or DHCP. It was a process by which each PC could ask for an IP address and the DHCP server would dole one out automatically. It still meant we

would have to touch every PC, but we just needed to enable the protocol on the PC and there was no need to assign an IP address to each one. This was something that could be done at any time before the cutover to routing. A sub-team formed, and resources started quietly turning on DHCP on all PCs attached to the token ring.

I worked with the Cisco guys, another part of the team worked with the Cabletron folks, and we put together the bill of material that we thought the final solution might look like. This was also called a Rough Order of Magnitude. This gave us a ballpark on the price tag of the overall project. This seemed like a never-ending exercise. It was during this time, when speaking with Steve, the Cisco SE, he asked me why we wanted to keep the 16-megabit backbone rings. They each had around 20 to 30, 4-megabit, local rings attached. I asked him what else could I use to make the situation better. His answer was FDDI. FDDI? Tell me more.

FDDI was Fiber Distributed Data Interface. It was a ring topology that operated at 100 megabits per second, and it would be a much better solution for connecting the local rings. I knew nothing about it, but I trusted my Cisco SE, he was way smarter than me. I knew I would never get approval for this solution so I did what anyone else would do, I added it to the BOM and didn't tell anyone. There were a lot of

moving parts to the overall solution, and nobody even questioned the new additional items on the BOM. The Cisco C7000's would just show up with the FDDI modules installed.

We made our formal presentation to upper management, and they seemed to be open to our proposal. The price tag was a couple million. Naturally, they wanted several meetings to hash out a lot of the details like where would we start, and how would we implement the solution. The project was approved and George named the team the Bear Skinners. The spotlight was now shining on us, and if it did not go well, none of us would be bear skinners for long. It was during this time, I started keeping my resume up to date.

The enormity of the project was something I never gave much thought to. It was going to take a while. We enrolled facilities management to help in removing all the Token Ring 8228s and adding a healthy fiber optic cabling system in every wiring closet. After some time, the project was completed. The beaconing rings stopped. The broadcast traffic was greatly reduced. Life was starting to get back to normal. Still a couple of issues to work through. We had just spent another ton of money on the network, and it wasn't all rainbows and sunshine. Like I had suspected the 16 Megabit building backbones were struggling to keep up with the newfound demand on

the network. Now that it was functioning much better, word got out and more and more people started to consume its resources. One Friday, the end was near. Seemed everyone's phones were ringing. Something needed to be done about the network slowness, again! During this time, I had secretly gone to all the wiring closets and started connecting the fibers to the FDDI cards. I had the whole FDDI network configured but I had the ports on the routers physically shut down.

Kevin came into my cubical. Desperate look on his face, he was feeling the end was near. Everyone was looking for someone to blame. "Can you do anything about the slowness?" he asked. That is when I confided in him my secret FDDI solution. I will never forget the look of disappointment on his face. "You secretly purchased unproven technology, and now you want to turn it on? They will have both our asses for this." I knew from my long conversations with Steve, FDDI was used in many places around the world with no issue, It was a ring technology. Of course it would work. Without pause, systematically, I shut down the 16-meg backbone ring and enabled the FDDI. If I was going to go out, it was going to be with a bang.

I never seen Kevin so mad. Looking back, I guess I should have included him in my plan, but I knew it would never be approved and I would be back in front

of upper management asking for more money. We all knew how that would turn out.

A small commotion was happening in the hall, sounded like excitement. Someone was asking, "What did you do?" I heard people replying "nothing, I didn't do anything". That was the standard answer around here when asked that particular question. It got closer. Same conversation. The manager asked Kevin the same question, flustered, not knowing how to respond, he pointed the director to my cube. The director appeared in my cube doorway and asked me, "What did you do?"

This was one of those make or break it moments. I would either be a hero or a zero. I could just hear the folks in upper management saying, "The long hair freak did what?" I confidently replied with the words no one else had uttered, "What are you talking about?" "The network, it's faster! It's a like night and day difference, so I will ask you again, what did you do?" As casually as I could, I replied, "I turned on the FDDI ring." He looked like a computer trying to process a divide by zero error. Something they cannot do, and often blow up trying to. When the wheels in his head stopped spinning he replied, exasperated, nervously, "What FDDI ring?"

I spilled the beans, told him the whole plan, the conversations I had had with the vendor, Cisco, and

how I used the fiber plant to build the connections between the FDDI interfaces. At first he seemed like he was going to lose his cool. I told him I would shut it down, right now, right away. “You will do no such thing!”, he exclaimed. “It’s finally working, do not touch it!” With that, he stormed out.

It’s like the old saying goes, it’s easier to get forgiveness than permission. Eventually, I was forgiven for my transgression. After enabling the FDDI ports at the other sites, the transaction speed to the mainframe from all PCs was dramatically faster. Except for one last pesky issue. The WAN link to the building in Chandler Arizona, would go down every day. We would have to bounce the port on the router to get operational. Doing so caused a lot of sessions to crash and have to be restarted. There was never a shortage of catastrophes.

It wasn’t long, I was getting all the heat for this. The long-haired, freaky router guy. The Cisco routers were mainly purchased because of my recommendation. Once again, disgrace had fallen upon the department. I was in trouble. I called my Cisco SE, he said, “You know you can call John Chambers.” I jumped at the chance. Ring, ring, ring, on the far end of the phone call. “Hello, this is Rick Kauffman with Motorola” The assistant knew who I was, we had just made a substantial purchase. “I conveyed that I would do unnatural things with a dull knife to certain

parts of John's anatomy, unless he had his best Token Ring guy in my office, early the next morning. To his credit he did, they fixed the issue. Turned out to be a Motorola chip on the board, ouch!

A couple months later I signed up to go to Cisco Networkers in Dallas. It was at the convention center in the west end. You complete your registration for the event upon arrival, and you are given a name badge on a lanyard. You wear it around your neck for the whole event just so people will know who they are talking to. I had just finished the registration process and quickly sought out the restroom. I was standing at the urinal and as I took stock of the men's room décor, another gentleman came in through the door and took up a position to my right. I noticed his name badge said, "John Chambers"!

I finished by business, washed my hands and made my way out the door. I knew there was only one way out and I was waiting. This was going to be epic. John came out of the door, and I immediately stuck out my right hand and with a big smile on my face I called his name, "John!" We shook hands, he had no clue who I was. I saw his eyes trace their way to my name badge. He saw the name and suddenly the 'cutting with a dull knife' conversation must have popped into his mind. I will never forget the look on John's face; It was priceless. I assured him all was good and thanked

him for all of his support on the account. He left, anatomy intact.

Drunks and Fools

Networkers was the biggest party I had ever seen, and they knew how to do it right. Way more booze than you could ever drink and ample food from around the world. Networkers had a tradition of giving you a silly hat and asked that you wear it all night long. Every year a different silly hat, it was the only time it was worn.

A live band would be ripping, flashing lights, people on stilts, and a mermaid? No, sorry that was Chicago Networkers and a whole other story! The event was closely resembled a carnival atmosphere. I was in a strange time in my life, and I was drinking way more than I should. I remember I had a beer in both back pockets, another in my hand and I was desperately trying to find the men's room.

Ah, I found it. I made my way through the door and upon entry I noticed I was having a hard time finding the urinals. Nothing but doors. Just then, one of the doors opened, and I saw a middle-aged woman, smartly dressed, with a slight look of disappointment on her face. She said, "Looks like you're in the wrong

place”. My feet were frozen to the ground! As much as I wanted to, I felt I could not escape this embarrassing situation. I was locked in by her gaze.

Thankfully another woman entered from the hall and let out a laugh. It broke the spell I was under, and I was able to make my way out of the woman’s restroom. Right next door was where I wanted to go originally, without hesitation I opened the men’s room door and found sanctuary from my horrifically embarrassing situation. Moments later, greatly relieved, I decided I have had enough and I needed to get back the Adolphus Hotel.

Earlier that evening, I had taken the shuttle bus to get to the event. It seemed like it made a hundred stops to pick up people from all the other hotels in the area before arriving at the event. Felt like an eternity and I’ll be damned if I take it back to my hotel that was only a block or so away from the event. The evening was fair and the walk would sober me up, a little. At the event, I was given a cheap, plastic, see-through briefcase with papers relating to the event, I stashed the silly hat inside it. I had it slung over my shoulder, I disposed of the remaining beers I had. By disposed I mean I drank them. Dumping them out would have been akin to alcohol abuse in my book.

I made my way out of the event and got my bearings. This was the west end in Dallas. Famous for the

location of the assignation of President John F. Kennedy. It was a short walk from the Dallas Convention Center up to Commerce Street, that was the street my hotel was on. I would head up to Commerce and make a right. The hotel would be just ahead, on the left. So, I set out on my short journey. The walk up to Commerce Street was uneventful, I was humming a tune and enjoying the cool evening air. Things changed drastically when I rounded the corner.

This was back in the days when I had a longish mullet and a full beard. A regular mountain man. At six foot and one inch I was not small. I was wearing a tee shirt that had a huge alien face and with the see-through plastic briefcase, I was quite the sight. One must count one's blessings.

I was slow on the uptake. I was trying to comprehend what I was looking at. Up ahead, about a hundred yards was a McDonalds. I was surrounded by a huge mob of black people. They were shouting and making all kinds of commotion. They were not happy at all. Beyond the crowd, in the distance, I saw the shimmering neon sign of my hotel. I suppose I could have backtracked to the event and rode the shuttle, but I was drunk. They say that God watches out for drunks and fools, so I was doubly blessed.

I'm a happy-go-lucky kind of a guy and I get along will everyone. Somehow I don't think anyone in the group would believe anything I had to say along those lines. All I had to do is keep my head down and keep moving and I believed I would be all right. I put my feet in motion and headed straight for the crowd. The first thing I heard was someone say, "Who is this asshole?" The rest of the comments were of a similar nature. There was a lot of tension, they kept trying to get me to engage with them, but I just kept my head down and kept moving. The crowd parted like the Red Sea.

I think they couldn't believe some crazy white guy would have the temerity to even attempt such a stunt. As I made my way through the crowd the mood shifted to more of astonishment than anger. It was like they couldn't figure out what was happening. I kept saying to myself, just a few more feet, keep going! Finally, I was clear from the crowd, I was busy counting my blessings and checking for traffic as I crossed the street to the side my hotel was on.

I was about fifty feet from the front door of the hotel. I found myself walking behind a white couple, even in my drunken state I thought it was odd. I did not care because this guy in front of me was accompanied by a slender woman with a very short skirt and extremely long legs with six-inch pumps. A much more welcome sight than what I had just experienced. Twenty-five

feet to go to get to the hotel and this couple spun around and the gentleman shoved a shining silver handgun into my ribcage.

I looked at the gun; I looked at the guy. I looked at the gun; I looked at the girl. I was so confused I did not know whether to flip or fly. I was very slow in reacting to the situation. I think they were just as confused by my appearance. I finally took a hard look at the girl. Through my bleary eyes I finally sorted out that she was actually a man.

That's when the guy pulled the trigger on the gun. I heard a loud CLICK. No bang. Reactively I blurted out to the girl, "You're a Dude!" That's when she, rather he started crying uncontrollably. The guy tried to console her. I reached out both arms grabbed them both by their shoulders and made a path right between them.

I did not hesitate in making my way to the hotel entrance eager to avoid getting shot in the back. I made my way to the elevator and up to my room. I sat down on my bed and tried to piece together the events I experienced on the walk to my hotel. Nothing made sense. Did what I think happened, really happen? My head was spinning, the room was spinning and I started to wonder if this was one of those flashbacks my parents had warned me about. I looked up as if speaking directly to God and said,

“Looks like I owe you one” and then passed out with all my clothes on.

The morning came, my head was pounding and it was only day two of the event. I was in sad shape and I don't drink coffee. I popped an eight-hundred milligram Ibuprofen and washed it down with a Red Bull, the breakfast of champions, and I was ready for action. I took the shuttle back to the convention center and walked inside. I had talked to the lady at the information booth the day before and saw her at her post. She smiled as I approached. I struck up a conversation with her and told her I walked back to my hotel last night. A grave look came over her face. She said, “Do you know what happened last night at that MacDonalds?” I was about to reply with “Some crazy guy walked through the crowd?” and she quickly retorted with “They killed two police officers!” I thought to myself, I owe the big guy way more than just one. To this day is a debt I will never be able to repay.

Surfs Up!

Nearly eighteen years had passed since I started chopping the legs off transistors. My technical skills had grown more than I had hoped. I still was responsible for the box with a red bridge and that

meant I would have to work with Ron. Ron was an Analyst who had worked with Pete, the guy who I used to share a cube with. Ron had helped in setting up the Cisco CGS in the back room. He was an older gentleman, nearing retirement. I loved to sit with him and hear him talk about his days as a sys admin. I would regale him with stories about my wild nights in bars, playing in rock and roll bands, and drinking way more than my share of beer. Ron told me, “You’re a member of the pistol club then.” I was a bit confused. Pistol club? He reached up on his bookshelf and pulled down a beer stein. On it were the words “Drink ‘til dawn, pistol noon.” He handed it to me and said, “You deserve this much more than I do.” It was an act of kindness I will never forget. I will take that stein to the grave with me.

At Motorola we were responsible for our own training plan. Basically, there was a pile of money and we had to spend it. Naturally, I would request to go to every Cisco “Networkers” event. One day while looking at an industry news-mag, I saw something that said, “web world”. I had no clue what this was. I took a walk over to Ron’s cube and asked him if he knew anything about it. “Sure”, he said. “It’s a bunch of URLs that represent different resources on the Internet.” “THE WHAT? I replied. He went on to explain that a URL was a Uniform Resource Locator, and the Internet was the network mostly made up from the DARPA initiative. It’s a bunch of universities

connected to one another via routers. There was a company called Netscape who were developing an application to look at URLs. It might as well have been some other language he was speaking. His advice was to go the conference and learn everything I can.

My stock in the department had gone up. I learned that if I told Laura, the administrative assistant that booked the travel, she would do what you asked. If you leave it up to her, she'll find the closest red-roof inn. Not because she didn't care, she just didn't have the time to research a better location. I suspected she was getting kickbacks from the red roof inn. I gave her a piece of paper. The conference was being held in San Jose at an upscale hotel. I indicated I needed to stay where the convention was to save on the auto rental.

I took a cab from the airport, SFO, and settled into my plush hotel room equipped with an adjustable firmness mattress. The morning came, I loved the hotel, it had a restaurant that had cook-to-order breakfast. I loaded up, you never know when your next chance will be. Pancakes, Bacon, Hash Browns, and hot black tea to wash it all down with. Perfect, I'm ready for anything. Including a red bull to pour on top of all that caffeine to kick it into hyper drive.

I made my way to the convention center. I walked through all the vendor booths. I wished my name was Peter Parker. There was an overabundant use of the term web slingers. Seemed like it was web this, web that. It was obvious to me that spiders were somehow involved. At the end of the row of booths was a larger booth with a PC. The first thing I asked, when I arrived, was for the specs of the beautiful computer. It was a fine machine. Tall, slender. He said, "It's a 386 processor, with 4 meg of RAM and a 120 Megabit hard disk. VGA graphics card." I was impressed, I have never seen a machine this fast. On the screen was an application with a throbbing blue letter "N". A Netscape browser, a real one!

The guy who was working the booth told me a story about in the very near future, everyone will use a PC and a browser to access content on remote "web" servers. I could picture it, a giant network, like the one in Router School. Resources in every city around the world. A web of connections. The world wide web or www for short. Ok, got the concept. The booth attendant told me that the communication was built on HTTP, or Hypertext Transport Protocol. He went on by explaining how content is tagged and stored on the web server with a markup language. HTML, or Hypertext Markup Language. Every URL started with <http://www>. I understood that to be Hypertext Transport protocol, world wide web.

Once I got back to my office, it was budget time once again. I was asked what equipment I wanted for the next year. My answer was, I wanted five Sun Sparc 20 workstations. You just couldn't get a better, more powerful Unix machine. Ron had inspired me to try my hand at being a Unix sys admin, and I wanted to build a mini version of the world wide web. Not much of a justification was needed, and my wish was granted. In a few short months, I had my work cut out for me when the new systems arrived. I loaded httpd on all the systems, bought a book on programming HTML, and started building my web servers.

It wasn't long after that I had my first web page that resembled a bulletin board and an application for submitting change control requests. We used that application all the way up to the day I left Motorola. I often think of that honorable member of the pistol club, Ron. If it were not for him I may never became so interested in the web and how it worked. That knowledge would serve me well in my new adventures. My only regret, I could have registered any domain I could have thought of, ford.com, chevy.com, gold.com, etc., etc. Just one of plenty of missed opportunities. One thing I did learn, was that technology was starting to move at a faster pace.

A few months after that, I got an offer to interview for a pre-sales engineer at 3Com. I was getting bored at Motorola, but the money was good, but I needed to

be challenged. I went to the interview with my long hair and alien t-shirt. My resume was impressive, router this, web that. This was a customer facing role. I didn't look very customer friendly, and I did not get the position. The hiring team must have seen something they liked and within a week I got another call from 3Com.

I was contacted to see if I would like to interview for a post-sales engineering position. I have spent years working on the Datacom team. The terms pre-sales and post-sales were new to me. Sales? I didn't know the first thing about selling. Turns out post-sales is after the selling is finished and didn't require any selling knowledge. The person I interviewed with asked me how much I was making at Motorola. Once I told him my current salary, a small, sly smile came to his face. He told me that if I took the position, that the compensation would be twice as much! I was speechless, for the first time in a long time. I had been at Motorola for eighteen years, it was comfortable, and I was at the top of my game. Torpedoes be damned, since when has not knowing anything stopped me before? I was ready to learn more

He told me that the hiring supervisor would be calling to set up a job interview the following week. I told him that I would be on vacation but would gladly take the call. I interviewed with my future new supervisor, over

the phone. He had one remaining question. He asked me if I knew the difference between a VLAN and an ELAN. I guess I must have impressed him with my answer. “ELANs are emulated and VLANs are virtual. Same thing if you ask me, but what do I know? Not much. He told me I got the job. I would be leaving Motorola soon. The first thing I did when I got back to Phoenix was cut off all my hair and bought some business casual clothes.

The smell of money

It was my last few days at Motorola, out-sourcing had taken a stranglehold on the Information Technology team. Management would take individuals who were under performing, and farm them out to an outside company. This was called out-sourcing. They would be re-badged as that company's employee and sent back to Motorola to do their regular jobs.

To me it seemed cruel and non-personal. I didn't have to worry about that happening to me because I had given my two weeks' notice and was a short timer. I was on my way out, and the timing seemed perfect. The out-sourcers were basically task master's and if you didn't perform there, out you went. Like I said, harsh.

My new supervisor, Terry, called me. He had a couple of things he wanted to talk about. He said, “I know you're still finishing up at Motorola, but we will be having a “team meeting” at Lake Tahoe, and we would like you to fly up and meet the team. We get together every ninety days, with all the engineers from the territory, and I think it would be a perfect opportunity to meet them.” I asked my Motorola supervisor, and he was agreeable.

The second item of discussion was, he wanted me to pop over to the hospital on Twelfth and McDowell. I knew the facility, and from our prior call, I knew that would be the location I will be working as a Resident Engineer. Terry wanted me to meet the hospital IT staff. The meeting was cordial, and I thought they were all very nice people and it quelled some of my anxiety just by meeting with them. I didn't know it at the time, but we would become very close friends.

Travel for the trip to Lake Tahoe was booked for me, and I embarked on the journey to meet the “smart” guys and get my first glimpse of what I had gotten myself into. Once again in my career, I was starting a new adventure and feeling clueless.

I had never been to Lake Tahoe before. It was a very interesting place. I jumped in my rental car at the Reno Nevada airport and made the drive to the lake. I read that this was a very deep lake. It was just about

springtime and there were small patches of snow in shaded areas. The daytime temperatures were pleasant, but the nights were still very cold. As I drove the wintry, winding roads that were way too close to the edge of the water, I felt my grip on the wheel getting tighter. It was a tense situation as I thought I was about to go off the road and I would never be heard from again.

Thankfully I arrived at the hotel and my first thought was, “wow, it’s not a Red Roof Inn”. It was very upscale. I checked in and made it to my room. Something had told me that I had arrived, so to speak. I was in the big show. Almost like getting called up from the minor leagues. I was dressed in my new khaki Dockers, and an oxford button-down shirt with one of my new neckties, a depiction of a Picasso painting. I dropped my suitcase and headed to the bar.

I saw Terry, with a couple other guys at a table in the back of the room. When I approached I could tell I was the one who “Didn’t get the memo”. The other engineers on my team were dressed in blue jeans and company branded polos, very not business casual. My new team members were a fun bunch of guys. They loved to drink. I learned that at any future event, if I wanted to find anyone on my team, I should always check the bar first.

The formalities started the next morning. In a small conference room, we gathered and discussed the issues and challenges within all our accounts. The level of knowledge with this team was amazing. They were talking about things I had very limited knowledge of. It was your standard team call. If you had any gripes or complaints, this was the place to air them out. Someone was having issues with one thing or another, another engineer would pipe in and say they had the same issue, and this is how they fixed it. Lots of complaints about travel expense reporting (the bane of my existence, I would come to find)

This was a three-day event. During the meeting I was unusually quiet. Observing more than anything else. Watching each engineer and noting where their passions lie. Some were strong at routing, others we specialist in Asynchronous Transfer Mode (ATM) and wide-area protocols. After the meeting, it was back to the bar and then dinner, which came with more drinking. During the dinner, I sat next to a guy who was very amiable. We struck up a conversation and I said, "So we get to have these meetings for two more days? Seems like we got around to just about everything anyone wanted to talk about". He let out a small laugh and said, "No, meetings are over, now we're more into the team building exercises." Oh great, I was not aware of the team building agenda.

Turns out, every ninety days, the team meets up and a luxury resort for three days. It was called Offsite Planning. That is how it got past the bean counters. The next two days were wide open. Terry asked me, “What is it you want to do?” I didn’t grasp what he was asking. He told me I can go golfing, snowmobiling, ride horses, go sky diving. It was wide open. Anything within reason. A few of the other guys were into golfing and the NorthStar course was clear and available. I rented clubs, bought golf balls, a glove, and there was plenty of alcohol and expensive cigars. It was a lot of fun; I could not believe there were companies who had money like this to burn.

Then I remembered that every ninety days we would be doing this again, seemed like a great way to build comradery. I found out the next Offsite Planning meeting was on Catalina Island, where we will ride horses up and down the beach and party like rockstars all night long. I could not wait to start working for 3Com. Something told me it was going to be a wild ride.

Crash, Boom, Bang!

Token Ring networking had seen its day, as well as many other network topologies. There was a new sheriff in town, Ethernet. It was becoming popular

because it used CSMA/CD. This stood for Carrier Sense, Multiple Access with Collision Detection. In a nutshell, an ethernet station would wait until the wire was not being used to send packets. This is used in combination with collision detection, which means, a transmitting station detects collisions by sensing transmissions from other stations while it is transmitting a frame. When this collision condition is detected, the station stops transmitting that frame, transmits a jam signal, and then waits for a random time interval before trying to resend the frame. This sounds quite aggressive and was quite different from Token Ring.

It was around the same time we saw Novell, AppleTalk, and other vendor specific network operating systems and routing protocols give way to Internet Protocol, or simply IP. I always say, In the end it will be ethernet and IP. I am still amused to this day, that people still cling to EIGRP, a Cisco routing protocol. EIGRP could simultaneously route IP, IPX (Novell) and AppleTalk. Today IPX and AppleTalk are akin to peaking Latin. Novell had what I think was one of the best host addressing schemes. It would use a network number and add the end station layer two address. Simple. Still, it would be crazy for engineers to try and deploy something like the internet with Novell routing.

Joining 3Com at this time was one of my most important transformations. I needed to ramp up on this “ethernet stuff” as fast as I could. Now, this was the time when the Internet was in its infancy. We were still using Netscape browsers and search engines like Dogpile, I will wager you haven’t heard of that one.

I didn’t have the luxury of going to some training class to learn about this technology. It had to be done the hard way, by reading books. I picked up a copy of Radia Pearlman’s *Interconnections*, and read that until my eyes were red, tired, and burning. Thankfully, Radia sums up the whole book in a simple poem. The spanning tree. This was the introduction to another bane of my existence. A Spanning Tree recalculation was comparable to Token-Ring beaconing, albeit a self-correcting, shorter network outage.

Spanning Tree Protocol removes the redundant loops from an Ethernet network. It examines all the links and determines a loop free topology, known as the tree. If someone were to plug-in an Ethernet cable into two ports on the same switch, a loop would be created, and Spanning Tree would have to recalculate the tree once more. This would take 40 to 50 seconds. Let me tell you 40 to 50 seconds of network outage was an eternity.

It seemed to go without fail. Someone would unplug a device, from a switch in the data center. Then

someone else would come by and think they would be doing someone a favor by plugging the disconnected cable back into the network, without ever knowing the other end was still plugged in.
LOOP!

This would lead to telephones ringing, Network managers storming out of offices and 40 to 50 seconds of pretending like we knew where the root bridge was and what happened. I bet you don't even know where your root bridge is either. One Spanning Tree recalculation could wreak havoc with the server protocols. Sure, the network was only out 40 to 50 seconds but the loss of traffic forwarding on the network would leave indelible mistrust of the network department. Every single recalculation was a black eye.

Becoming a resident

I had a unique role at 3Com. It was called a Resident Engineer. That meant, that you lived at the customer's facilities. In their offices, forty hours a week, if not more. The entire network at all seven customer sites was built on 3com Routers, Switches, Core devices, and Wide Area Networking. As a first line of defense, the customer contracted with 3Com to have a 3Com engineer on-site during business hours. This customer was a giant healthcare organization, and

hospitals never close for the evening, so there was always someone assigned the duty pager.

My new manager, Terry, was the current resident engineer and he was just promoted to SE or Systems Engineer manager. He wanted to get out of the position just about as fast as I wanted into it. He was eager it get me “up to speed”. The quicker I onboarded, the sooner he could fully embrace his new role. A passing of the torch, per se.

There was a team of hospital engineers as well that I was going to have to work with. They all looked pretty much “buttoned down”. Khakis, oxford shirts, and a wide array of colorful neckties. I would have to work very closely with that team. I really didn’t know them, and they did not know me. It would be quite an adventure getting to know everyone. Turns out, this new team was a bunch of practical jokers and were very skilled at their practical joking craft.

There was an engineer that worked on the server team. He had a habit of saying “right” a lot. He was just a normal guy; I think he was from Cleveland or something like that. He said the word “right”, like he was Crocodile Dundee. Here’s how this little prank went. One of them recorded themselves, trying to mimic the guy’s voice, saying the word “right!”. They proceeded to make this recording the sound his computer played every time he received a new email.

People would notoriously walk away from the computer without locking it. This was a wide-open opportunity for the guys to make their move. Change the email notification and turn the computer volume all the way up. Every time the guy got an email, the computer would play the wav file. “RIGHT!” would reverberate through the office. It was a coordinated attack. All the guys waited until the perfect time and started filling up the guy’s inbox. I laughed so hard I cried.

In another episode, they took packing tape and stretched it across the opening of another guy’s cubicle. Top to bottom. An invisible wall. Then someone else came with giant bags of shipping foam peanuts and filled the cube. Turns out this was frowned upon by upper management and the jokes became more refined. Being the new guy, I knew it wasn’t long until they could cook up a scheme aimed at me.

Every Friday morning, I would leave for work early. I had a few extra stops to make. I stopped at the doughnut shop and picked up a dozen, then a stop at a bagel shop to pick up bagels for the guys who didn’t want all the sugar in the doughnuts. I walked in the office and saw the guys were waiting eagerly for my arrival. I placed the treats on a table in the break

room and after grabbing a chocolate frosted, made my way to my cube.

Honestly it wasn't a cube but only a corner where no one else would sit. Right in the front window. Like being on sale at Macy's. I thought it was odd, it looked like the guys were following me. Why weren't they gorging on the doughnuts? I rounded the corner and to my astonishment, they replaced my desk with a wooden shipping pallet. Someone had taken a picture of my desk. Torn it completely apart and reassembled it perfectly, down to the last detail. A very impressive attempt. I had a feeling like I was finally accepted and a sense of honor for them to plan such a scheme. It was quite the effort, but you know what they say about karma.

We would turn our computers off at night. In the morning, it would take an eternity to get it powered up, get the applications loaded and get the vpn working. Frequently, I would just get all this working and one of the guys would call me over to their desk. When I returned to desk, I would discover that my computer was turned off. I would hear snickering from beyond the cube walls. Foiled again.

So, I brought in a couple extension cords from home and came to work very early. I ran the extension cords under the cube walls and connected several of the guy's computers to them. Then, I plugged the end of

the extension cord into an outlet under my desk. I would wait until all the guys got their PCs fired up. Then I would wait a few minutes more. I would pull the plug under my desk. Oh, the sound of confusion was so comforting. Again, I would wait a minute or two and plug it back in. Clickety-click, I could hear them starting to get the PCs back online. A minute later, pull the plug. I did this about three or four times until one of the guy's discovered the root cause of the power outages. PAYBACK!. They were all impressed, I could not only take their abuse, but I could give it right back. I became part of the team, and we developed lifelong relationships that last to this very day.

The ATM Doctor is in

My “internship” was over. My hiring manager, Terry, who had been with me for my first week, had to let me fly on my own. He just wanted to be sure I could do the job. The last order of business was to find out why the wide-area networks were systematically crashing. The connections between the hospitals were built on a product called the “Path-Builder”. The whole hospital system was connected in a giant “T3” ring which ran at 45 megabits, 44.736 to be precise. The ring was a Sonet ring and was built on ATM or Asynchronous Transfer Mode. ATM was a cell-based

technology, and it had far more acronyms than the military. I should know because I was in the Air Force, that's where this story started.

Naturally this was all new to me, so I went into my "fake it, till you make it" mode. Terry asked me if I knew anything about sniffers. A sniffer was a protocol analyzer that you could connect to a circuit or a network and see what's on the wire. I had formal Network General sniffer training, and I was confident with my "sniffing" skills. Just to add some context, a sniffer was built on a "luggable" PC, with a very specific configuration. A unique network interface card was added with the Network General software. The price tag was around ten thousand dollars. Sniffers were hard to get, and you could only keep them for a limited time before they needed to be shipped to the next site. Today you can download Wireshark and get the basically the same thing for free.

Terry told me he had ordered an ATM sniffer, and it was going to be shipped to a hospital in the east valley. I met him on site, and we inserted the sniffer into the wide-area network link between the main site and the one we were in. Breaking the line to make this connection created an outage and we coordinated it with all parties before doing so.

My new office was a wiring closet where I would sit on a chair and watch the sniffer until the outage happened. With my manager beside me, I watched the sniffer screen and rubbed my chin and made noises like I recognized something. I could not help but feel the watchful eyes of my manager. He was not familiar with sniffers and had no formal training. So, it fell upon my ‘advanced sniffer training’ to solve the problem. I was starting to feel the pressure. Fortunately, it didn’t take too long before the whole thing went belly up.

Whatever it was, had just happened and I could see the look in Terry’s eyes like “OK, Rick, tell me what the problem is”. I studied the screen. I did not even know what I was looking at, but I didn’t want to let the cat out of the bag. I examined the sniffer trace more closely. I noticed a pattern just before the system crash. It was something called ICMP. Internet Control Message Protocol.

If you wanted to check if a device was responsive on the wire, you would issue a “ping’ command. The result of typing ping 10.10.10.1, would result in the message getting wrapped up in ICMP and sent to the desired address. If the device was online, it would respond. A very simple way of testing network reachability and used by network management systems to keep track of whether a device is up or down.

I told Terry “I think I see the culprit”. The look on his face was astonishment. He was excited, happy, and euphoric. Talk about pressure. I pointed to the screen and said “This IP address pings that IP address about 50 times and then the path-builder goes offline. I just don’t know where that IP address is”. The look on his face changed to something like he had just seen a ghost. He knew exactly where the offending IP address was. It was on the computer sitting back on his old desk at the main site.

Terry was a good networker, and he knew that he needed some form of network management to keep an eye on this far-flung customer network. Network management tools were expensive, and they were always the last thing most people thought of. Sad. Terry had found a “free” tool that makes a map of the network and sends ping, or ICMP, commands to network devices to determine if they are on or offline. A harmless piece of software.

It turns out, that the path-builder had a bug in its code. It was intermittent and would not happen all the time. The bug was related to the management interface of the path-builder. Turns out, under very specific conditions, if you tried to ping it, it would ping you back with a vengeance. So much so, that all the CPU on the path-builder would be consumed and the device would just quit working. The temporary fix was

to reset the management card. Later, new code was created, and a patch was applied. Problem solved.

Unfortunately, we had to have a “Mea Culpa” meeting with the hospitals directors and admit it was a 3Com bug, and the outage was created by a tool deployed on their network, by 3Com. Ouch. The meeting went better than expected. The directors were very happy the “new guy” found the problem, and this went a long way to earning their trust. They thought I was an ATM guru. Why would I let them think differently. One last outage was scheduled as we had to disconnect the sniffer and send it on to the next customer outage.

Hammertime

M.C was the initials of the network director’s name. Just about the time that rapper MC Hammer was popular. It was unavoidable, the guys on the team started calling him Hammertime. He was an amiable guy and was pleasant to work with. In the beginning of my residency, I didn’t interact with him very often. As time passed, we developed a mutual respect and that went a long way to making me feel like an integral part of the team. My manager, Terry, was moving on to a different new role and 3Com was looking for a new System Engineer manager. It was an excellent

opportunity for me to follow in Terry's footsteps, but I had my fill of managing people when I worked for Motorola, and I knew enough to steer clear of that opportunity.

Hammertime called me into his office, he wanted to talk about something. I was thinking "Oh great, what have I done now". I went to his office, and he said, "Close the door please" He didn't seem too upset, and that piqued my curiosity. He asked, "What would you think about me taking Terry's position?" I gave it a moment and the first thought in my head was better you than me. He liked management, I did not.

I replied, "I think you would be an excellent SE manager". I could see the conflict on his face. He was a long-time employee of the hospital and was struggling with the decision to take a new role. He said, "What I mean is, you wouldn't have any concerns with me becoming your new manager, would you?". "Absolutely not!" was my response. From what I had experienced he was fair and not a micromanager.

It wasn't long, the transformation was complete, Terry out, Hammertime was in. One of the hospital's analysts, in another department, was promoted to Network director, his name was Mike. I didn't know him. His appearance was a bit alarming. He had a military flat top haircut, one of the best I had ever

seen. He looked like a Marine drill sergeant. I wasn't quite sure how we would get along. Turns out he was a great guy. Looks can be deceiving. The gang would get together and go to lunch just about every day, and he would always try to join, schedule permitting. More on Mike later.

The hospital was planning a 12-million-dollar network upgrade. It was the Y2K flap. People thought the world was going to stop working because computers at the time had only two digits for the date. The New Year was approaching, and the date was about to roll back to two zeros. Oh, the humanity. It was amazing the catastrophes that people believed would happen and never did..

The word came down to upgrade everything. It was great news for 3Com, better news for the 3Com sales team. The first thing to get upgraded was the core. New ATM chassis switches were acquired. ATM uplinks and 100MB Ethernet ports, the latest marvel of technology. The hospital's lead network engineer came up with a rather ingenious architecture call the "H" bone. Looked like a great idea, seemed like it would be very resilient. We went forward with it.

The first hospital was upgraded, all new switching in the wiring closets. The rollout went smoothly, well-coordinated. Only the equipment for one hospital

was procured. Upon successful deployment, funds would be released to acquire the other locations.

We were just about to start with the high fiving each other and a call came into the helpdesk. It was from the newly upgraded hospital. The look of disappointment on everyone's face meant that all funds were on hold until we could figure this out. This was very concerning to 3Com, especially from the sales guy's perspective. We drove out to the site with the issue and interviewed the person who called it in. We asked if she could show us what the issue was, she could not, everything was working perfectly.

We waited for quite a bit and there really was nothing wrong. We looked at each other with dumb looks on our faces, shrugged our shoulders and drove back downtown. It wasn't long, another call came in from that facility, The first question was, "Is it the same user?" The answer was no and that was not the answer we wanted to hear. Turns out, it was a different user, with the exact same issue. We drove back to the scene of the crime.

We interviewed the new user, same issue, and same as the last call, everything was ok. This was a mystery, in just a few minutes time, we got two more calls, from the same facility. I was starting to feel the panic, the problem would pop up and disappear just as fast, a professional game of whack-a-mole.

I knew the drill, I needed a sniffer, this time an Ethernet sniffer. I needed to plug into the ethernet switches in the wiring closet. I set up shop in a large wiring closet, I had no chair so I flipped over a trash can and started what would become a six-week effort. Every time the problem occurred it was on one of the other switches where I didn't have my sniffer plugged in. One day I was watching the trace and one of the user's called out across the office and I immediately stopped the sniffer.

Whatever it was, I finally had a trace of it, all we had to do was interpret it. It appeared to be a layer two issue. One moment the workstation's layer two address (MAC) was on one side of the H bone, then the user's workstation would crash. Then the layer two address appeared on the other side of the H Bone, What? I had sat in that wiring closet for six weeks, on an upside-down trash can, where I played the game of cat and mouse with that outage. Now we were trying to puzzle out what was going on.

Hammertime was getting the heat, it was coming down from sales, hard. His job was on the line. They were worried the rest of the revenue was going to slip, if not vanish. I saw Hammertime looking very worried. I saw the customer looking angry. It was a very tense time. You could say it was hammering time, no

question about it. If this did not get fixed, we were out.

We all huddled in a conference room. The team scoured the sniffer trace, the inventor of the H bone let out long “hmmmmm”, he saw a problem. The first glimmer of hope was forming. It appeared that the MAC address was jumping back and forth across the H-bone. On one side everything was fine, the MAC would jump to the other side of the “H” and the traffic to the user’s PC would stop.

This time it wasn’t something that 3Com had connected to the network. It wasn’t a 3Com bug. Once we disabled the ports on the core switch, the ones that connected the “H” together, the problem was resolved. It was a team effort, but for some reason I was recognized for sitting on the trash can, for six weeks. Without that effort, I doubt we would have ever captured the problem on the sniffer.

With Hammertime being previously employed by the hospital, it went a long way toward repairing the relationship between the hospital and 3Com. After a meeting with Hammertime, his manager and the hospital CIO, the project was back on schedule. The money started to flow, and the other facilities were scheduled for a replacement.

Mike's mystery

The new network director, Mike, turns out to be a huge second amendment guy. He loved guns, even had one he would pull on a trailer hitch! He told me one day he had hundreds of guns. He was a walking gun encyclopedia, a book full of knowledge for you young folks who don't know what that is. We had long conversations about my days in the USAF and all the guns I got to train on. I had a few guns myself.

The day came where there was a new network outage. We were stumped, as usual, what the issue was. It was early enough in the day, we should have it ironed out quickly. We did not. As the day waned close to four PM, I noticed more concern on Mike's face than usual. He was worried big time about something. I had to know, so I asked him. "Everything all right Mike?"

He looked at me and said "I'm in some deep trouble. It's my birthday and my wife has dinner planned out; we're going to some fancy restaurant." "Well, happy birthday!", I proclaimed. He nodded but the worry persisted. The rest of the guys gathered and told him not to worry; we'll stay as long as it takes to get the network back up and running. He was already running late and out the door he flew!

I had to get on an airplane the following day and travel to Dallas Texas for the 3Com annual Engineering training event. I had just made it to my prison cell, I mean hotel room, when my cell phone rang. I answered it with my usual proclamation, “RICK!” The news was devastating. It seems Mike hastily drove home, late for his birthday dinner plans with his wife. For some unknown reason, he told his wife he needed to clean a gun before they left. Mind you, they were already late for his birthday, and his wife was a bit heated.

I don’t know about you, but I think my wife would beat me over the head with the gun before I could clean it. He proceeded to clean the gun, and it went off, putting a hole right through his head. Mike was no longer among the living. I sat on the edge of my hotel room bed and I was numb. Nothing about that story made any sense. He was an expert handling guns, why would he insist on cleaning the gun? It could wait. When he left us at work the previous day, his main concern was making to dinner on time. The whole thing was quite sad and tragic and to this day does pass the smell test.

The beginning of the end

In the days following, 3Com made the brilliant decision to get out of the networking business. They

were ending all of their Corebuilder, Netbuilder, Pathbuilder, and Linkbuilder products. Sent a letter to all of their customers to move their business to Extreme, not Cisco. It was devastating news to the network team; the look of disappointment was on everyone's faces. All of the stuff we just installed in the network was about to be replaced. More on this story in a minute.

3Com set up an Engineering transition plan. Every engineer would be given a stay on bonus. You stay with the company until your date; you get the bonus. They fled like rats leaving a sinking ship. The other thing 3Com offered was online Cisco training. I completed a Cisco course every week. As the other engineers fled the company, I stayed and took their training budget. I studied as hard as I could. Like my life depended on it. This was pre-CCIE days and I got seven Cisco Certificates, and I displayed them proudly in my cube.

The command came down from on high and the networking team needed to pick a new vendor. I had worked myself into the team as a valuable resource. I was able to accompany them to the meetings they had with each vendor. Nortel, Cisco, Brocade, there were a bunch of them at the time. I really wanted them to choose Cisco as the vendor. I knew it well, and I was convinced it was the better choice for the hospital.

I was completely amazed as we left the Cisco office — they had totally blown it.

The presentation did not go over well, and they missed on a lot of targets. It was a done deal; they were going to move forward with Nortel! I could not believe it. I knew the guys at Cisco well from my days at Motorola. I called up one of my old sales guys and told them they need to get all hands-on desk and ask for a second chance. I expressed the concerns I heard from the team and told them to reach out for a second chance. They did.

After the dust had settled, Cisco was going to be the new vendor. I felt strongly that was the right decision. During this tumultuous time in the industry, the networking landscape changed beyond all recognition and Cisco was one of the only vendors to remain seemingly unfazed. I basically handed that account to what had become my arch nemesis and would spend many years trying to get that account back.

The hospital was paying 3Com a large amount of money to have me sit in their office. If they were transitioning to a different vendor, they didn't need me. Looks like my time was coming to an end, and my livelihood at 3Com. I had one thing going in my favor. My seven Cisco certificates. None of the guys on the

network team knew how to work on Cisco switches. I had a meeting with the director of the IT department. He told me he was not renewing my contract, and I needed to start packing my cubicle. I told him that I was still a valuable resource as I could personally train the team on how to work with Cisco infrastructure. It sounded like a good idea to him and my contract was renewed for another year, made the sales guy happy and got me past my stay-on bonus date. At the end of that year, we had all of the core networks rebuilt and most of the hospitals converted to Cisco networking.

3Com had ventured into a new technology called Voice Over IP or VOIP. I knew nothing about Telephony. I was about to learn more than I ever wanted. I had reached the end of my final contract with the hospital, and it was a melancholy departure. I still stay in touch with some of those guys to this day. I was ready to learn something new.

Mr. Watson, come here!

Coincidentally, 3Com did not lay me off. I survived and thrived installing telephone systems all over the Southwestern region. The solution, 3Com NBX, a voice over IP system. It would take you voice and wrap it up in ethernet frames and send it all over the world. Amazing. It was a box about the size of a

microwave oven. It did all the processing. There were adapter cards that connected to analog phone lines, and T1 circuits. In the beginning I was in over my head. The best way to learn a product is to go somewhere and install it, fight through all the installation issues, get it up and running. I was in a different city every week installing this solution. I learned every trick that box had. That year I had fifty-two Avis car rentals. That is when I really learned to hate traveling for business.

At the end of that first year out of the hospital, I became one of the best installation guys, I could make that solution do unnatural things and the customers were ecstatic with the results. The only bad part of the installation is when you look at two hundred tiny boxes, each one containing a phone that needed to be assembled. I felt as if I could train a monkey to do that part, but I would have spent more time taking care of the monkey and getting him on the plane than it was worth.

The absolute worst part was I would have to talk to two hundred people and ask, "Where on your desk do you want this phone?" It would always be where the six-foot patch cable would not reach. Then I would have to explain, "It needs to be here, that is as far as the cable reaches" The look of disappointment on their faces was undeniable. They would always retort with "I don't like my phone there"

Then one day I got a call. I needed to fly over to Dallas Texas, Love Field. I flew a lot on Southwest Airlines. Turns out the Dallas Cowboys bought the NBX phone system for Texas Stadium in Irving Texas and they were having a lot of issues with the product. They were very hot and threatening to call their lawyers. I made my way the stadium. I told the receptionist who I was, and I was whisked away to Mr. Jones office. I never saw a customer so mad.

He cussed me out; it was the most profound chewing out I have ever lived through. I told him I will do my best to make things right. I am the guy who wears the white hat, and I ride a white horse. I guess the analogy was understood and I went about trying to make things operate as designed. It took me eight weeks and a lot of sleuthing to fix all the issues.

I met with the network manager, and he showed me where the NBX was plugged in and racked. I was amazed by what I saw. The old phone system, a big Nortel PBX was hooked up to redundant power sources and multiple rows of batteries. I played dumb and asked the manager what all those batteries were for. He proudly proclaimed that they were necessary to insure continuous operation of the old phone system. Then he showed me the 3Com NBX. It had a single power connection to a wall outlet. No UPS (Uninterrupted Power Supply), no

backup battery banks. I waited for a few ticks until it dawned on the manager that the new phone system would need something more reliable as far as power connections. I could see how embarrassed he was, so I let it go, point made. Then I asked for diagrams of the network connections. None existed. The look of disappointment on my face was understood as well.

It was the company who sold the solution to the Cowboys who really did not do themselves any favors. There are well specified requirements for deploying VOIP. You must have ethernet switches in the network. They did not know what I meant by the word switch. It was all hub, shared, technology. I finished my tour with the Cowboys and got them into a stable operative state. Over the course of many visits to the Cowboys, I became good friends with the network manager, furthered my career and had plenty of new stories to share with my fellow engineers.

I had one more call to Texas, it was a small town I can barely remember. Its downtown square looked like it was right out of Mayberry. One of my final NBX installs. I was installing the phone system for the city. Courts, Jail, and even 911. It was very interesting. I returned to my hotel after the first day. I asked the guy at the front desk where the closest place was to get a beer. He gave me an awkward look and said, "This is a dry county". I was taken aback! He said, "Don't worry,

I know a place where you can get a beer”. “Awesome, do tell”, I replied.

“Well, first you got to drive out this county road until you get to a fork in the road, take the road to the left and keep going until you see a small trailer park on your right”. I kept nodding my head making mental notes. “You walk up to the blue trailer, it will be on your left and knock on the door, they guy will ask you what you want, you give him some money, and he will bring out what you need”. I had the thought to look around to see if there were hidden cameras and I was in some kind of a prank TV show, but no, it was all real. Undeterred, out the door I went and got in my rental car and drove the directions. Sure enough, there was the fork in the road, the trailer park with a blue trailer, I approached and knocked on the door.

A man answered, nervously looked around, and asked what I wanted. I figured I would not want to make this trip every day. I ordered a 12-pack of Bud. He told me how much money he required, I gave it to him and he disappeared. Moments later, he reappeared with a twelve of Bud, and it was very cold. I turned and headed for my rental car with my newfound booty, and the man shouted and said, “Do you want to drink a couple of those beers at the topless bar?” I was dumbstruck for a moment and I replied, “Where is it?”. He pointed to his left and said, “That yellow single wide over there”. I didn’t skip a

beat, “Oh, look at the time, it’s getting late and I really have to go!” head down I walked briskly to the rental and drove away wondering what would have happened if I said yes.

The days seemed to go on and on. The money was good; the job was survivable. I kept my head down and kept on doing my thing. Hoping one day to get back to the world of big iron networking. Little did I know my dream was being worked on. In backrooms around the corporation, the game was afoot.

There was so few 3Com engineers left, I was literally covering the western half of the US. I believe, at this time, there were only twelve of us pre-sales SEs in the entire company. It made me think of the time, early in my 3Com career, I was in Salt Lake City at the 3Com building. It was like a theme park. Vivid colors, millions of things with blinking lights in every cube. The people were very, very smart and I was in awe just being worthy enough to be invited to the party. In the end, out of all those amazing people, I was the last 3Com engineer covering the whole state of Utah. I had outlasted them all, including the building. The Gods were smiling on me, it was the first time I had the thought, just maybe I can really do this job.

The return of the router

Hammertime took a different role in the company, and I went to work for a new manager. I enjoyed working for and with him. He always challenged me, which forced me to grow my engineering skills. One day he called me and was excited to share something in private. He said we, 3Com, was getting back in the networking game.

I did not share his excitement; we had some low end networking products and knew we could not compete in the data center with Cisco. He told me a story about a partnership with another switch vendor, and we were going to have switches the size of refrigerators! I was taken aback, somewhat in disbelief. Was it true?

The “other switch vendor was Huawei, a Chinese based company. They were the “Cisco“ of China and 3Com was going to partner with them in a new company called H3C. H was for Huawei, and 3C for 3Com. Partnership with a new juggernaut in China gave 3Com sellers a whole new story to tell. And the best news of all? They had a configuration interface which was almost identical to Cisco. Game on!

Cisco won the telephony wars as we pivoted back to selling big ethernet solutions. The VCX (big brother to the NBX) was backhauling a ton of IP traffic for

providers, but Cisco's marketing and engineering were relentless. They also had a way of making it hard to meet customer expectations for E-911. They could deliver the first responders to the individual's desk. Competitors could only get to the right office floor. We were starting to pick up some traction and had some massive customers interested in our products again.

Around this time, I was moved into the Federal sales team, and my life was upended once again. I had to think in terms of agencies and departments; I went from post sales to pre-sales. I had to become more of a sales minded person. I was in the perfect place to do it. Surrounded by some of the best Account Executives who ever played the game. I learned quick. There were a million sales tactics like North and South, Keeping up with the Jones, and SPIN. It was like I was in the ultimate sales academy. I learned all that I could.

From these seasoned sales professionals I learned the three rules of selling. One, nobody will buy anything from you if they don't know who you are. Two, nobody will buy anything from you if they don't like you. And three, you will not be successful unless you know why the customer needs what you are selling. What it is has nothing to do with it. Everybody has a what and they make the mistake, when talking

to customers, to waste all their time explaining what it is they're selling and not why they need it.

The only reason they wanted me on the Federal team was, I was in relatively close proximity to the US Army installation where they did all the vendor network testing. The test facility was in the south-eastern part of Arizona. All the other members of the team were in Washington D.C., and travel budgets were tight. It was a three-hour drive for me to get to Sierra Vista AZ, home to Fort Huachuca, I know what you're thinking. How in the hell do you pronounce that! Think of it as WAH-CHEW-KA. I got to know it well. Home of the TIC and JITIC testing centers. It's where they had put together the gauntlet of network system testing.

Each vendor was given a three-week period to build the network architecture proposed by the US Army and pass numerous tests. A lot of vendors did not pass. They're put back on the schedule, months later to try again. The whole time not being able to sell into the Department of Defense. It was a solid blow to the bottom end of any organization trying to sell networking to Uncle Sam.

I was told the switch vendor before me failed, and lobbied congress to change the testing requirements. They had rolled in with semi-trucks full of gear and a hand full of engineers. I pulled up in my company

Ford Escape with my laptop bag and a head full of confidence.

I had no clue what was expected, this was my first time for any of this. The gear I had shipped there was early versions of the H3C solution, and it was neatly stacked on pallets waiting for me. It had all the basics, OSPF, BGP, and Layer two VLANS. I could not wait to get started. The TIC engineers were jolly older engineers and were very welcoming and accommodating. I was going to be there for three weeks, and I either had to drive the six-hour round trip every day or stay in a local hotel. I opted for the hotel right next to the Outback. I dug into the assignment with the mindset that this was no big deal, some VLANS, a little routing, some aggregated links, easy.

The TIC engineers shared with me the network diagram, the core, distribution, and access layer, I have deployed this configuration a million times. Layer two throughout and routing in the core. I could do it in my sleep. The real test was the multicast test. It was brutally amazing and really hammered the gear hard. I couldn't wait to give it a go. It was early Friday morning of the first week and I approached the TIC engineers and said I am ready for the test. Those guys looked at me like I had three eyeballs! "No way!". "Are you sure?" echoed throughout the testing lab. My reply was "There is really nothing left for me to configure."

I did not know most vendors, with a team of people, took two full weeks just to get set up. I guess my lack of knowing how things worked was a huge benefit. I handed the system over for testing and drove home to have a well-deserved weekend.

I drove back down on Monday, through the Main Gate and made it back to the TIC. I struck up a conversation with one of the engineers who told me he was just starting the Multicast testing. I played with Multicast before and I wasn't too confident the new gear would make it past the onslaught of testing. It was really quite impressive. It was almost like they wanted it to break. It did not. It passed that part of the test. The TIC engineers looked amused.

It was like that all week. More testing, more passing. It was late Thursday and one of the engineers told me everything looked good. The excitement inside me grew. Then he said "There is only one more test. You need to programmatically change one of the VLAN names, without using the command line. This was a setback.

I went back to my hotel that night and worked on a plan. This was before the days of REST APIs. I knew the network devices supported SNMP, or Simple Network Management Protocol. If you could come up with the correct string of numbers, like

1.3.6.1.4.1.9.1.451 you could manipulate the value assigned to the OID (Object Identifier).

I found a SNMP manager application on the Internet and after a very long download, it was on my laptop. I quickly figured out how to use the tool. The next day, I plugged into the network and started snmpwalking. A process of scanning all of the OID strings. I stumbled upon the values for the VLAN name. A simple SNMP Set command and I proved to myself it could be done. I met with the TIC engineer and demonstrated the solution. He watched, somewhat amused, rubbing his chin. I could see the wheels turning in his head. He gave a short nod and said, “It’s a bit of a stretch but I’ll accept it”.

The news quickly spread through the lab. 3Com passed the TIC testing in two weeks, on the first try. I called my manager and told him the news. He was blown away. At one time in the past, he was the engineer trying to get the networking gear through the TIC test and the equipment failed on the first attempt. The good news was celebrated. There were plenty of high fives and smiles from everyone. I tore down the gear and put it back on the pallets for shipping. We were now able to sell into the Department of Defense. It wasn’t a couple months later, and I was involved in competing for a deal with a huge Air Force base. We won the deal! I was making quite a name for myself.

Education is the key to success

My days as a Federal Pre-Sales Systems Engineer were behind me. I was moved onto a State, Local Government and Education (SLED) sales team covering the four corners region. Again, this was a new environment for me, I had never worked in the education sector. I had a lot of new things to learn. The only cities who were buying lots of 3com, back in those days, were Albuquerque New Mexico and Salt Lake City Utah. I spent a lot of time in both of these cities and worked with some very large school districts. Because I dealt with State and Local Government, one of my customers was Los Alamos National Labs!

I was so excited to go to Los Alamos, one of the most tightly secured areas in the USA. It was a long drive from Albuquerque, past Santa Fe, and on up the mountain. It was secluded that's for sure. I had heard so many stories about the place, I couldn't believe I was going to see it with my own eyes. In the end, it was just like driving around any other small town. We met the team over in the networking group and they showed us around (as much as they could). I got to see a real Cray-1. It was the early days of big data and super computers. I could only stare in amazement. I love technology, especially very expensive

technology. I would make numerous trips to Los Alamos and got to sell a lot of switches to some very fine people.

If I wasn't in New Mexico, I was in Utah. So, first things first, if you are going to navigate Salt Lake City you need to learn the grid. It all starts with the Mormon Temple downtown. Ground zero, if I may. Then the roads were numbered, with larger numbers as you moved away from it. I quickly learned how to get around. I don't know why, but I was surprised by some of the 'locals' on the downtown streets. They had some very sketchy characters.

It wasn't long and I was navigating the city like I have lived there many years. I got to work with a great bunch of customers; business was always hopping in SLC. One of our resellers had an engineer who I worked with every time I was in town. A young man, very pleasant. He was a Mormon. I was the polar opposite. I spent my weekends in a life of debauchery, and we would exchange stories of our adventures each time we met. I got the feeling he was living vicariously through me while I was learning a lot about Mormonism from him. Nice enough group of people but not my kind of thing.

Back in Albuquerque, at a public school, they wanted to deploy this new technology called wireless networking. In other words, WIFI. Wireless Fidelity.

The technology was new and mostly compared to a black art. The school wanted four thousand access points. I didn't even know what it was or how it worked. I said, "four thousand, that shouldn't be a problem". That night I was on the web, learning as much about it as I could. Seems like that was all I could do. The technology was moving so fast, it was very difficult to keep up.

I had to do a walk-through of all the schools and figure out how to place the access points. I also had to design the network that would support all of it. I had no tools to analyze coverage. All I had was my imagination. Needless to say, the deployment fell short of what the desired outcome was. I met with the network manager, and I will never forget the look of disappointment on his face. As technology progressed, WiFi analyzers were starting to become available. I acquired one and I was able to clean up quite a few issues. Because of the size of the deployment, everyone thought I was a WIFI expert. I wasn't, but I wasn't going to tell anyone. That was another one of the secrets to my success.

We just introduced a new product. It was the 3Com Intelligent Management Center, or IMC for short. It was a full F.A.C.A.P.S. network management system for wired and wireless networking and it was amazing. The only downside was a steep learning curve. Another secret to my success is to find the

latest technology, one that few people knew about, and learn all I can. Turns out if you make a simple PowerPoint deck with a bunch of stolen slides. Present it like you know what you're talking about, and people just assume you're an expert in that subject.

I started using IMC and I quickly went from a command line jockey to someone who can use a powerful software solution to tame wild networks. One of the biggest things that really help owners of massive networks is configuration backup. Just that alone was worth the money. IMC did that and many, many other things. I was just starting to understand how this product was going to impact my future.

Big changes in a very short time

I was early to the game with IMC. I knew how much of a game changer this platform was. It would literally manage Cisco products better than CiscoWorks. I saw this at several of the customer accounts I'd call on. Wall to wall Cisco switches and they were still trying to manage everything with the CLI (Command line interface). IMC was an awesome Trojan Horse; I could bring a lot of value to those customers if they would just buy something other than Cisco. I planned to sell it as much as I could. It pulled through a lot of hardware sales.

I recall one day I was in Utah finishing up some business and looking forward to going home. I walked onto the plane, working for 3com. I turned off my phone and started reading a book. Most likely some Neal Stephenson, thousand-page tome. I am a huge fan and carried those monstrous books every time I traveled. The plane touched down in Phoenix an hour and a half later. I turned on my phone and a notification popped up on my news feed. 3com had been purchased by HP, Hewlett-Packard. The first word that came to mind was 'invent'.

The game changed just that fast. I felt a sense of pride, HP was a very big player, and I had just made it to the big show! HP was hiring more engineers. It was a very exciting time. I was promoted to a consulting systems engineer, or CSE. I would parachute into problematic accounts and try to pull a win out of something that was starting to look like a loss. I was good at working with upset customers.

IMC had a module that provided NAC or Network Access Control functionality. In short, it would determine if you were allowed on the network, where you could go, and what applications you were authorized to use. I must admit, it was very difficult to figure out how to install it and make it work. I spent many days in my home office, toiling, and I became an expert on the product.

The rage of the day was B.O.Y.D. or bring your own device. Didn't matter what device you were using, it was identified and connected to the authorized network. This was the hottest technology. A lot of customers wanted it. There was a big customer in the Bay area. 3Com, or should I say HP, sold them IMC, and NAC. A pair of engineers were dispatched onsite, and they began working on installing the software.

The first things they did was download the manual and start reading. They struggled and did this in front of the customer. It was late in day five, a full week of trying with nothing to show for it. I could not believe it. Back then, HP farmed a lot of the services installations out to third-party contractors.

Monday morning my phone was blowing up. It was my people leader, no longer allowed to call them managers. He gave me the dump on what was going on and wanted me to get involved. The customer was threatening to call their lawyers. I can't remember what it was, but I was unable to travel. I told Brett, my people leader, I can do it all remotely, all I needed was access.

It took me a couple hours to survey the damage. It was a mess; the basic IMC install was not up to my standards. It was real simple stuff too. I could not let it stand. It would eventually lead to issues in the

future. I told the customer I was going to delete everything that the two engineers did in forty hours. Start over from scratch. He was flabbergasted! In a very hostile voice he asked, “How long will it take you to make it work!”. I stayed calm, you catch more flies with honey than you do grease! “I think I can have it operational by tomorrow morning”. I confessed.

I think that even made him angrier. Why was he charged 40 hours for something I could do in less than a day. Good question. My people leader was taken off guard. I was not in the same state as he was, but I could just imagine the look of disappointment on his face. I was taking a huge risk. “Are you sure you can get this done by tomorrow?” he demanded. Full of confidence as usual, I reassured him that I could do it. He was right, I was sticking my neck out, but it was nothing that I have not done at least twenty times. I got access and I got cracking.

I worked until one in the morning. It was functioning as designed. I went the extra distance and made them a custom landing page. The customer’s attitude was completely different. He could not believe it. He asked what I did to make it work. I explained how it functioned and demonstrated multiple devices accessing the wireless network. Disaster avoided. Another happy customer. My stock in the company was going up!

Jedi Knights and APIs

By this time, I was very fluent with IMC. I knew how to make most of the dozen or so modules work. It was like magic. I was deep into it. I used to go to presentations where I joined with the godfather of IMC, Les. He was a tall, very well built, bald headed, black man. I swear, the first time I saw him, I thought he was Michael Jordan. He was very gregarious and he could present better than anyone. I just watched and learned. There was a hand full of engineers who really picked up IMC and would champion it all over the globe. Les was a huge Star Wars fan, who wasn't, and he started calling us his Jedi Knights.

One day at a team meeting at our offices in Marlborough Mass, we are in a conference room just joking around and one of the guys, Chris, another huge Star Wars fan, started saying, "If I'm a Jedi Knight, then I want to be Darth Vader". I was a good bit taller than Chris and I had this big beard and longish hair. "Who am I going to be" looking for suggestions from the group. Chris paused a beat and he said "Chewy!" Ever since that day, I became the Wookie. I run www.wookieware.com, and www.techworldwookie.com. My license plate on my truck says wookie2 (I'm a Wookie too!), technically not a Jedi Knight but I liked the name and it stuck.

One day, Chris called, and asked if I had ever heard of REST APIs, or (Application Program Interface). REST meant that they were representational of the communication state. I did not know anything about them. That was very short lived. He challenged me to write a python script. I had years of coding Fortran, COBOL, and Rexx. One programming language's basic operations are similar to others. I thought how hard could it be? I needed to pick up some Python chops, fast! I was excited to learn something new. At this point in time, the internet was full of useful videos, and you could learn almost anything you want on YouTube and Udemy. It wasn't long I knew enough to be dangerous.

The request was to build a script that would find the physical address of a device or host on the network. A simple web front end, input box for the address and a button to click to kick off the script. It was during this time I started picking up a lot of skills which allowed me to start my journey to becoming a full-stack web developer. A dream of mine since the first web world conference I attended so many years ago.

The task became a lot more difficult than I ever imagined. I was trying to use urllib3, a python package for talking to APIs. It was like learning how to talk to Martians. I persisted. I put up a picture of Thomas Edison on the wall in front of my desk. When it felt like I was never going to figure it out, seeing

Edison, the man who knew ten thousand ways how not to make a light bulb, gave me the inspiration to keep going.

Eventually it worked. The code was ugly as sin, but it worked. I called this script Columbo, named after the TV detective, and I was hooked. I loved Python and eventually I started writing ten thousand lines a year. This wasn't my job, I was a networker, but it was something I could point at and say, "I did this, let me show you how it works". Again, this was something very few people knew anything about, so it just added to the persona I was building my whole career. I started using requests, another python package, to work with the APIs and it was much easier. I dug in and started learning all the APIs inside IMC. It was akin to witchcraft, it was magical. A new way to deploy networks without the command line or web interface.

It was around this time that we started selling Arista switches. We were selling BigSwitch, as well as H3C, Procurve, and a white box switch that basically ran open-source Linux routing software. Five different switching solutions. If you tore the sheet metal off the top of each one, inside they were all identical. Strange times in the Kingdom indeed. It all came down to how you could build networks and take advantage of the software stack. That is when it hit

me. I need to specialize in automation, and I can be gainfully employed for quite a while.

I was working on a deal with a software company who wrote games for the iPhone. They had to deploy 8 racks of servers for each new game. There were two switches in the top of each rack. These top of rack switches have connections to four spine switches. This network architecture was cleverly called spine and leaf. The issue at hand was the engineers could not deploy the spine/leaf quick enough. They were taking a couple of weeks to configure all the routing and the complex protocols involved in the spine/leaf architecture. Arista uses something called CloudVision for managing their switches. I was quite surprised to find out it had an API.

I started developing a solution to reduce the time it took to deploy the network. I built a demo lab with a virtual machine that I could run CloudVision and virtual instances of Arista switches. I wasn't long until I had a working solution. I used Jinja2 template to build snippets of the switch configuration. The application took inventory of the links between the switches. I stored them in a Mongo database. Those links were analyzed to create the configuration snippet. From initial deployment (addresses on the switches) to a fully operational spine/leaf in less than five minutes! I called this application 'Antigua'. Just one of the numerous open-source applications I have

developed. I showed it to Arista management and the first thing they did was try and offer me a job. I had one of those, what I wanted was five hundred thousand dollars (never got it). This just demonstrates the power of automation. I was on to something big.

Containerized

It's one thing to write a cool application that runs seamlessly on your laptop, try giving it to someone else. Big trouble. If your friend's computer does not have the same python packages, it will never run. This was the driving force behind DEVOPS, a pipeline that increased the velocity in which applications were developed, tested, and made operational.

The invention of Docker containers, among others, made it possible to develop an application in a “virtual” container, and anyone who had docker installed on their system could easily run your application. Once I discovered this technology, I would use it for every application I developed.

The next thing I discovered was something called docker-compose. A simple deployment tool that let you write in a text file, called a YAML (YAML Ain't Markup Language) file. In the YAML file you could describe how many containers to spin up, you could

have database containers and define all the networking between the applications containers. All you had to do to get the whole thing up was type `docker-compose up`. Want to tear it down? `Docker-compose down`. Not only did this make my application portable, something that could run on any docker machine, but it turned my infrastructure into code. A simple text file.

It was just the thing to start building open-source applications that a partner or a customer could easily download and run. The company I worked for was not only a network vendor, but they also sold servers, data center management applications, and lots and lots of data storage systems. Turns out every one of those products have its own API! The thoughts in my head were racing. I envisioned all of the devices in the data center, rich with telemetry, just waiting for someone to come along and ask for some of it. You could conceivably pull information from a server and use it to configure parameters on the network. Or get information for the storage array and kick off another process on any number of devices.

My head started to swirl. I needed something in the middle of it all. Acting like a traffic cop. Looking for a certain condition to exist and when it happens, turn around and execute actions on something else. I was foolish enough to think I could code something that would do the trick. That was a crazy thought.

I eventually stumbled upon Stackstorm. It was just what I needed. It is very sophisticated in how it operates but I was able to pick up the basics quickly. Stackstorm had sensors that could be defined to look for just about any event. When the sensor was true, it would load a trigger that would call a rule, that rule was defined with workflows that would call a series of actions. These actions are just basically python scripts that call APIs.

Stackstorm had a series of action packs already written for SolarWinds, Twitter, Cisco, there were a lot of them. Just waiting to be consumed by the Stackstorm framework. I set out to write integration packs for all of the devices we sold. I bundled these together in an open-source project called the Digital Services Engine. It was just a collection of Stackstorm integration packs. I built a cool demo. Twitter was hitting the mainstream. I saw the Stackstorm had a Twitter integration pack on the exchange. I downloaded it and integrated it into my Stackstorm server. Kind of like installing apps on your phone. You want Twitter, you just download the application and the next thing you know, you are tweeting.

I started thinking about the Fabric Composer, a tool we use to deploy spine/leaf networks. I created an integration pack for that. I wrote a Twitter sensor.

Every time I would send a tweet, the Stackstorm Twitter sensor would trip, and a Twitter action would collect the tweet message body. In the body was the word VLAN followed by a number, let's say the number 12. In the rule list of Stackstorm actions, I would call the Fabric Compose action to create the VLAN on my data center fabric. How cool was that? I learned an important lesson at this time. Don't develop solutions that nobody cares about. Stackstorm is still a viable automation platform today. How long it will last? Who knows, like I said before, technology moves fast.

DCA to the rescue

My time as a network systems engineer was coming to an end. A new experimental team was being created. I was invited to join the team. The Data Center Architects. This team was composed by server, storage, and networking engineers. All working together on the full data center stack. We were all experts in our field as a primary skill. We were challenged to come up to speed on the other products we knew very little of. The goal was to empower a group of data center systems engineers. People who knew compute, storage, and networking. Along with that a whole host of software applications.

It was a crazy time, so much more to learn. I would pitch to a customer about data center networking

one day, and the next day I'd be talking about data fabrics, and cloud to another. I was soaking it up. It was only natural; I would visit some very large customers talking about their network and in the next breath we would be talking about RPOs and RTOs (Recovery Point Objectives and Recovery Time Objective). Important things to people buying storage.

I was working with a local Account Representative, and we cracked open a huge deal. A couple million in networking and a whole lot more in compute and storage. People still looked at me like I was the network guy and, in the end, that's what I needed to be in this deal. Next deal, different role.

After digging through the proposed architecture, I knew I was going to build four separate data center fabrics. Connectivity from one fabric to the other involved Palo Alto Firewalls. The customer wanted two-hundred and fifty VLANs between each fabric. Not a problem on the network. A single range command and it was done. The Palo Altos? Not so easy.

Using the web interface on the firewall was very cumbersome. It took an extremely long time to click through all the options just to build one VLAN on the firewall. The customer knew this would not be accomplished quickly and it threatened to extend their deployment cycle. They had 24 firewalls to

program. The look of disappointment on the customers face was unmistakable. It was going to take a long time, and he was the engineer who had to program them.

I knew what needed to be done. APIs to the rescue. I got on AWS and spun up my own Palo Alto firewall virtual machine. I also created an Ubuntu workstation to write some code on. I built an automation tool called Panix, or Palo Alto Network Interface eXecutor. I am not the most knowledgeable marketing person but that's the name I came up with. Panix could program all of their firewalls, with all the VLANs, in about twenty minutes. If you know how to code and automate, these solutions are quickly prototyped and tested. In the end the whole DEVOPS things actually did a lot to increase my productivity.

Things were going well with the project. There was a whole team of people I had never met. One such person was Jim, an EA or Enterprise Architect. The EA had engineering control and acted like a quarterback, handling all the technical communications between the team and the customer. That guy knew quite a lot about servers and storage. I watched his every move. I was back in school — I wanted to be an EA. It wasn't long and Jim came to me and said, "I am looking at leaving the company, I have been offered a position where I can make some great contributions". I was dumbstruck for a beat or two. Then I said, "Are they

going to replace you?”. He replied “Yes, my people leader’s name is Todd G.” “That’s crazy!”, I said with excitement. I had known and worked with Todd at 3Com for many years. I called him and said I would like to join his team and he agreed.

Data, the only thing that matters

Once again, I found myself in a new role. Thanks to my time in the DCA team, I felt I was ready for the challenge. I joined the weekly Enterprise Architect call, and the things they discussed on that call made me feel like I was in over my head. These guys were some hard-core compute and storage engineers. I knew then I was going to have to really pick up my game. So, I went back to my tried-and-true method of finding something that the team knew little of, some bleeding edge stuff and start talking about that. There was no shortage of subjects to choose from. The company was buying technology at a frenzied pace.

I started with something I knew fairly well. Docker. Specifically, how Docker was used in the DEVOPS pipeline. I could always rely on my coding skills and knowledge of APIs to make me stand out from the others. I went back to my open-source project, the Data Services Engine. I had a whole bunch of new products, and I needed to start learning their APIs.

During this effort, I started to realize that the most important thing in a data center is the data!

It was data, specifically Business Intelligence and Data Warehousing that seemed to be what everyone wanted. The more data you can collect the better off your analytics were. They would want data from a lot of disparate sources. Finance, Customer Relationship Management (CRM) tools, Operations data, and a lot of other sources.

In order to make any sense out of this information, and perform queries on it, you had to get the information into a Data Warehouse. The process involved a data engineer, who would extract, transform, and load (ETL) the data into a common format to be consumed by the Online Analytical Processing (OLAP) engine.

The current process is a forty-year-old technology. Tried and true but most of all trusted by huge corporations to understand their business. Technology never stops advancing. It seems it was over night Data Warehousing started to have some real challenges.

It was hard to append data, real time operations were difficult, and meta data growth was in terabytes. The smart people in the room came up with a new

approach. The Data Lake! Something new that not a lot of people were talking about it.

In the beginning, the data lake had its own set of issues. It had poor support for BI; it was complex and started being called the 'Data Swamp'. Up until now Business Intelligence was more like a bunch of spreadsheets. Rows and columns of data. What we would call structured data. Emails, MP4 videos, voicemails, and chats had value as well and needed to be analyzed and mined.

These data sources were known as unstructured data. The promise of the data lake was it would store structured and unstructured data in a common format called Parquet. The promise of the data lake was, companies could use their legacy analytic tools on the data and data engineers and data scientists could start working on Artificial Intelligence.

As technology progressed, we saw the invention of the Lake House. The Lake House could ingest all the Data Warehouse information as well as all the information in the Data Lake. It had a Structured Transactional Layer, thanks to an open-source project called Delta Lake. There was no need to maintain multiple systems, gave you one source of truth, and it added ACID, which guarantees database transactions. Adoption was somewhat lack luster. Not a lot of people thought it was a good idea to

empty out their Business Intelligence Data Warehouses on new-fangled technology. I really couldn't blame them.

Then came a couple open-source projects like Ampool and PrestoSQL. They wanted to upset the apple cart. They came up with the concept of Distributed SQL queries. This meant corporations could leave their data where it was, safe and sound, and make all the remote SQL Queries from a centrally located server. In the end most companies adopted multiple approaches on how they mine their data for value. Turns out Artificial Intelligence will start to play a very important part in all of this, but more on that subject later.

The great cloud migration

Virtualizing workloads was all the rage. VMware was huge! Seemed every customer I talked to was using it, but there was a hidden tax if you did. You had to hire a bunch of smart people to operate the solution. This meant you had to secure and maintain a bunch of hardware & software solutions. It was an expensive hit to operating costs. Some other smart people started thinking, what if we could run the virtualization for customers. We could stand it up in a data center with a big connection to the Internet and

sell compute and storage to the masses. The cloud was born. Soon the Cloud Wars followed. All “Cloud” really means is you using someone else’s computer. It’s not in the sky, but most likely somewhere in a fortified data center in Virginia.

Companies were clamoring for compute and storage. They wanted it now, but most companies' Information Technology (IT) departments could not move that fast. If a line of business came to IT and asked for the resources a project required, a whole number of things would happen. The people who ran the storage group would immediately send out for a vendor bake off. The three most popular vendors would be asked to join in a proof of concept (POC).

The gear would be acquired, installed at the customer location, ran through its paces, and the performance and features could be verified. Of course, the other vendors would go through the same thing. After all the POCs were complete, a winner was chosen. It only took about 6 months to make this happen. Then someone would make a request to the network silo, and it would take an additional two weeks to configure the proper VLANs on the network.

This was a deathblow to lines of business trying to accomplish what was demanded from the customer. They needed to move much faster, so they pulled out the department credit card and paid for all of their

compute and storage in the cloud. This created a shadow IT organization and caused a lot of issues for companies knowing where all their assets were. Going to the cloud would only work on about thirty percent of the organization's applications. There were a million reasons why. Hard coded network addresses in applications, data security and data sovereignty to name a few.

The cloud infrastructure was fast and easy. You basically logged into a portal. You clicked on resources from a menu that was easy to navigate, choose how much and what type of storage you needed and most of the time, the network just knew what to do all by itself.

Compare that experience to working in the confines of the IT department. The public cloud had value. Applications like ServiceNow and Sales Force were public cloud resources that most customers used. It was easily consumed and fulfilled this need in companies who required it. You would never want to run them in a private data center.

What was to be done about the seventy percent of applications you could not run in the public cloud? The answer was surprisingly simple. Private clouds. Customers could get the same hardware and software stack that the public cloud used and run it in their back room. Same sleek portal, easy ordering

menu. Virtual and containerized workloads when you needed it. Often these solutions could be operated by the vendors selling them, all the way up to the workload. That meant you did not need a bunch of smart people to maintain the solution infrastructure. It was the best of both worlds for your workloads.

I became very skilled at evangelizing these solutions. One thing that never changes is change itself. I was working for a people leader that had no clue what the two Enterprise Architects on his team did. He reluctantly inherited us and we reluctantly accepted. The team calls were an endless discussion of weekend sports and talk of the old days. I never felt anything productive was discussed. It was a weird mix of storage, compute, networking and Enterprise Architects. Something happened like it always does and the company needed to shred some jobs.

The call always come on the Monday closest to October fifteenth. I was joking on the phone with a fellow engineer, “Hey, if the boss calls, don’t answer”. We would laugh. As I was talking, another call started to come in from my people leader. “Joe, it’s the boss, I need to drop, talk to you later”. I answered the incoming call. I didn’t have a care in the world. I just answered like it was any other day. It was not. He told me I had two-weeks and was eligible for eleven weeks of pay. Ridiculous! I had worked in this

company for twenty-five years. It was over. Out the door in fourteen days.

I didn't know whether to flip or fly. I was in a panic and I was scared. Like I said, I had twenty-five years and I wanted to retire from this company. I needed to get busy. It was due to all the relationships I had curated over the years that gave me opportunity. I emailed everyone I could think of. Telling my story. Looking for anything. I didn't care what the job was, I would learn it and be the best I could be. The next day I got an email from one of those long-time relationships. I worked with him at 3com many years ago. He was hiring and offered me a spot in a learning and development role. I gratefully accepted.

The difference between a student and a teacher

The new job was in enablement. I was to take the product information and disseminate it through the engineering ranks. This was accomplished by holding technical meetings where we would discuss detail product information and how to technically sell it. In addition to the technical sessions, I created physical hardware labs where engineers could get their hands on the actual equipment. I was out of sales, it felt

weird, but I got used to it fast. No more quarterly sales forecasting.

The data center networking labs were just recently brought online. There were ten pods of gear. They would start out in their initial configuration. The lab guide was used to configure and build the solution. The lab would take about four hours to complete. After the last engineer had finished, it was time to reset all the lab pods back to their initial configuration. I would have to manually log into each resource and reset them. The network controllers were the worst. I would basically log in and undo every command that was issued during the workshop. The whole process took about two hours to complete.

This was highly annoying and needed immediate attention. I have automated switching hardware for years, but when I started this job, I knew very little about VMware automation. I had a lot to learn; way more than I ever wanted. I started with one of my Docker applications. One container for the APP, and the other for a database. The process was mapped out. A script that would roll back the switch configuration, easy. During the course the student engineers built virtual distributed switches in VMware. I had to programmatically delete them. Down the rabbit hole I went.

There was a big trade show scheduled in Las Vegas in two weeks and the data center pods were to be utilized as much as possible. My people leader wanted to book the labs one after another. I told her the reset takes two hours to reset which limited the number of times in a day we could run the lab. One lab and a reset session would kill six hours. I could see the look of disappointment on my people leader's face. I needed to get this reset tool running as soon as possible. Another complicating factor was I would be in speaking engagements when the lab needed reset. The tool had to be simple to use so anyone could reset the lab.

I was head down developing the solution. The VMware reset was not easy. Little victories. Once I figured out how to clean off the virtual switches, in order to delete them, I had to figure out how to revert the VMware snapshots of the workloads back to the INITIAL snapshot. That turned out fairly easy to do. It was coming down to the wire, and I started working nights to finish the project. The extra effort paid off, with very little time to spare. In the end the reset took twenty minutes, and we were able to get two full crews through the workshop every day of the show. In three days, we touched sixty sales engineers.

Back home from the big show, the next issue to be resolved was the actual lab guide. The first guide was written in Microsoft Word. Simple enough, it would be

saved as a PDF file and disseminated to the instructor of the workshop. With frequent updates to the lab guides, this led to everyone having a different version of the guide. Something would change, which impacted how the lab was completed, and the instructors would show up to the workshop, with the outdated guide and not be able to complete it.

Once again, my programmer, DEVOPS, automation specialist mind went to work. I was familiar with markdown; a few of my developer friends had discussed it with me. You can write in a specialized way to indicate headings, paragraph's, lists, and tables. You could even insert pictures with captions. It was everything I needed for a lab guide. The hardest part was creating the first one. Completing all the markdown, I needed a way to convert it to hypertext markup language (HTML), something a web browser could easily read and display. I used Jupyter Book to make the conversion. Then it was very easy to store the results in Github, an online site for saving development projects and implementing version control.

There is so much more to do. I want to streamline how we interact with documentation. I want to develop a document flow for projects. It would be great to see everyone's To-Do list and see where I might be able to assist. Some sort of a portal where everyone would be aware of each other tasks and

action items. It would be nice to have a place where we could track the development of workshops. I see the road up ahead and it is wide open for me to run as fast as I can. If only I could develop faster, and adopt new technology at the speed of light.

AI ate my homework

I have mentioned earlier how I would write ten thousand lines of code a year, I did it just because it was fun to do. It was something that I could point at and say, “I did that!” Not many of my fellow engineers could. I wore it like a badge of honor. No matter what the challenge was, I could try a figure out a way to solve it with code. There is a saying “Any workload you can document, you should automate” I saw that on the back of an old Stackstorm T-shirt. Many times, I took it to heart and that is just what I did. I would code to solve problems.

That’s when tools like ChatGPT came bursting on the scene. It was a super nova of Artificial Intelligence. I was working on another VMware script. This one was a hairy one. I had to programmatically build a pair of virtual distributed switches for each lab. I thought, “Let’s see if ChatGPT can code this for me?” It did. I had a feeling of excitement when it wrote the code in about two minutes — I looked on in amazement. The

possibilities were exploding in my head. I downloaded the script and it didn't work.

I had to debug the code, which was much easier than trying to develop it myself. It didn't take too long, and I had it up and running. Back to ChatGPT I went. I was going to build a series of python functions, sort of like a toolbox for dealing with VMware. It was too easy and ChatGPT kept getting better, less time troubleshooting. It felt like I was cheating but I was getting a lot of work done. There was never a shortage of that.

After a while, I started to get really depressed. Something I worked so hard to obtain was gone. Now any kid in junior high school could write ten thousand lines of Python a year. The look of disappointment upon my face was visible on most all our team calls. I felt like I was robbed, and it was a personal loss for me. AI had created my sadness, but fortunately, another AI took the sadness away. In the beginning of this book, I mentioned that I play guitar. AI in music was going off the rails. It started opening up possibilities I would have never been able to enjoy, unless I had millions of dollars.

I have recorded over one hundred and fifty songs. I am not the worlds greatest singer or recording engineer. Using new AI tools like suno.ai and onemoreshot.ai made it possible to release music

that sounds like the best producers in the world worked on it. With onemoreshot.ai, I can make full length videos, all AI generated for thirty dollars. This was huge for me, and it really open my eyes up to what AI can do for all of us. With my sentiment toward AI realigned, I started thinking about coding again. That is when I first met Claude.

I thought ChatGPT was good, but Claude was different. It was more conversational. It was weird. I was developing an almost personal relationship with Claude, then I had to reboot my laptop. When I got Claude running again, it had forgotten who I was and everything we had discussed in the previous session. The look of disappointment spread across my face.

Bummer, it was like losing a friend. When I started chatting with Claude again, the first request was to remember me and our conversations. I also told Claude “Whenever I ask you what time it is, always respond with its 2:15 in New York.” Sort of an authentication statement. Now, when I log into Claude from my ipad, I’ll ask what time it is. When I see the answer about 2:15 in New York, I will know it’s the ‘real’ Claude. At least the Claude that has some ‘memory’ of me.

I have so many ideas for applications and I can get Claude to write all the code for me. I started ‘vibe’ coding with Claude. Claude might be incredibly fast

in writing code, but it has no idea what code to write. So far. I'm still needed in the application development cycle. I'm still a cog in the DEVOPS pipeline. After overcoming the high of asking Claude to build a swagger interface into my application and embed itself as an AI assistant, I'm starting to realize what I am witnessing. The future. It won't be long until Claude doesn't need me. It will be able to dream up its own applications. Then it will have the same problem I have. Trying to get anyone to use its apps.

I'm six months away from retiring. The end of a very long career. When I first thought of retirement, I thought I would wait until I'm seventy. Then people I have worked with for a long time started to retire and leave. I thought about what I would do with my time, everything I know will no longer be needed. Would I spend the remainder of my years just sitting around and watching the wheel of fortune? All good questions that I used to think I needed to answer. Not anymore. As the final working days pass, time slows down. Six months, easy, right?

I wonder what will happen when the end gets here. I'm afraid everything I do will be forgotten the minute I leave. I've seen a hand full of friends leave before me and they are never spoken of again. Time moves on technology is constantly changing. I heard someone say they expect ninety percent unemployment by

2030. I doubt it will take that long. The world will be a different place.

The singularity is near! If you chart the advancements in the personal computer, their capabilities are always better than the last model. What is important here is they are exponentially better. If you plotted this exponential growth on a chart, eventually the line would go straight up. That's where we were headed. Straight up.

In the next five years we will see a proliferation of AI and robots. I believe the robots will find their first commercial success being used around the house. Sort of like everybody putting Alexa in their home. They will be pervasive. They will have Claude inside their heads, and it will all seem perfectly normal.

One day, hopefully a long time from now, in my twilight years, when I ask my robot nurse what time it is, it just might respond with, "It's 2:15 in New York."

