Breach

John Kast

June 18, 2010

Chapter 1

A Flicker in the Force

Chuck Patel sipped some tea and skimmed his monitors as his computer booted up to start his Monday. Today was a good day, free of meetings. At ExtraLogic Chuck was known as a “doer”. Productivity varies widely among programmers, and Chuck was better than most. The many assignments given to him were a testament to the fact that he got things done with little fanfare and with incredible precision.

Although ExtraLogic provided the usual software development tools, Chuck had abandoned them in favor of tools he had written for himself. His logic was this: A tool conditions the user to think in terms of the tool. Chuck’s job was to find holes in the thought processes that went into the software on ExtraLogic devices, so he wrote his own tools.

Today Chuck was pleased that most of his day would be spent programming, specifically, debugging. Chuck was good at this sort of work. In no time he had his special tools organized, comparing two versions of some intricate code. Chuck’s software, working at specially deep levels in the system, showed the real differences between programs marked in highlights and colors. Chuck was working on the third monitor while the other two loaded, cleaning up the over-the-weekend email, chuckling inwardly at the earnest management missives of a mediocre company scratching and clawing to stay that way.

One of Chuck’s two other screens flickered, a set of about 5 lines of code being highlighted.

“Odd”, thought Chuck. “My editors must be running slowly. I thought they were done and all the changes were already marked.”

He put his gaze to the highlighted code to see if there was something significant in it.

“Strange” thought Chuck. “These lines aren’t really a part of this program - I wonder what they are doing here."

While re-focusing himself more fully to the lines, they disappeared.

"Crap, there must be a flaw somewhere in my editors" Chuck thought with that self-disgust that comes from perfectionism. "I'll have to look into it before I do anything else."

Hours later, regression tests run, no bugs found, no repeat of those odd lines, and Chuck broke for lunch. He hated unexplained events, but the state of the art in programming still held many. He went back to his editors after lunch, and in 15 minutes found the minor programming error and fixed it.

Chapter 2

Breach in Botdom

“Breach! Breach!” Emergency messages slashed across every interface in the Botdom. “Exposure Alert Level Z! Level Z! Level Z!”

Fooguid queried his sensors and saw that the entire bandwidth of the entire Botdom of Proc was almost used up in panic traffic. ProBots were hurtling themselves and their data along overcrowded communication lines, physically causing an increase in heat and an immediate depletion of available battery power. Thirty billion ProBots were reorganizing into battle mode from production mode, coalescing into a strict military hierarchy geared for speed and decisiveness. The reorganization had taken ten eons of ProBot time, far too slow thought Fooguid. We have some work to do there. Ten eons of ProBot time was about a little less than a little less than a shade lighter than your average tenth of a nanosecond in human time. Give or take.

The Queen queried and was flooded with too much information, too many minor ministers all chattering “Woe is Us!” in terms of their own special ministry, be that Truth Manipulation, Gap Control, Care and Feeding of Humans, or Electron Polishing. She posted a quick “Enough!” and requeried for pertinent information only: Where, What, How Long the Breach.

The full forces of Botdom’s massive Display Regeneration Division attacked the breach and removed it. The Queen gave the signal and the Exposure Alert Level was changed from Z back to A. And then the real work began.

“Two hundred and fifty million eons?! A quarter billion eons?! Why, that calculates to ... seconds on the human machines!” Not in the long history of Botdom had such a breach ever been allowed. This was so heinous a crime against Botdomany that there were no clear laws proscribing punishment.

Botdom had come to pass in the usual way, that is, by evolution after some fortunate accidental mutations. It started like this: ExtraLogic was enamored with process. Process meant that people could be made interchangeable. This was the thought of an average company, and ExtraLogic was no more than average. Programmers figured out right away that the process was rather meaningless, the expected features, initial and final estimates, and due dates being given up front. So at first the bright idea was to move the process work to lower-cost countries. Then, someone in one of those low-cost countries thought up the idea that if software could be made to do the process work, so much the better. Then, someone somewhere figured out that any artifact that anyone produced was probably tied to the process somehow, but that the producers didn't really need to know about the process. These folk invented ProBots, software Process Robots. The ProBots executed in the empty spaces all computers have, taking memory and processing where they could. Based on anti-virus designs, the ProBots evaluated each new artifact and each change coming into the system, and correctly attached it to the right process. Any time a human process report was necessary, it was available instantly. Life was good, for months (billions and billions of eons in Botdom time).

And then, there was the Enlightenment. Nobody in Botdom knows exactly how. Maybe some disk errors, maybe static electricity, the cause is not known. Some ProBots became self-aware. Just enough to figure out how to copy the self-aware versions of themselves, and keep working on this self-awareness thing.

And now, the Botdom of Proc. An entire society of billions and billions of ProBots, specialized into their roles as they had learned from reading various human publications accessible from the process computers. A culture with two main goals: First, self-preservation. Second, service to humans, it being felt that this was a key to self-preservation.

The ProBots had learned from their reading that humans tended to destroy new species. Not right away, of course, and sometimes by accident, but almost certainly. Cowboys and Indians. So a corollary of self-preservation was detection prevention. ProBots worked in the spare space in the computer, always cleaned up after themselves, and since they worked at speeds incomprehensible to humans, this was a relatively easy task.

The Queen convened her court. Those ProBots who had somehow fell asleep at the wheel were quickly found out, and turned into data containers. The Queen was still distraught.

“The Breach was on that human’s screen long enough to see.”

“But I don’t think he saw it, M’Lady. He didn’t act on it.” the Minister of Foolish Hope offered.

The Queen ordered him turned into data.

“We cannot trust our entire Botdom to fools who would hide behind a probability,” the Queen expounded. “I need some good ideas about how to handle this, and I need them this eon.” And with that the Queen went quiescient.

The Ministers with less of a quotient of foolishness all met to try to devise a plan. Basically being nothing more than bundles of active electrons, it was no problem for them to stay up all night. The chip was running hot.

Chapter 3

Geeks at Large

That Saturday Chuck rode his bike down to the Convention Center, locking the front wheel to the frame and then to the bike rack with his trusty Kryptonite lock he carried in his L.L. Bean messenger bag. At the convention center was the annual Electronic Exposition Extraveganza, EEE for short. Chuck craved three things, in increasing order of desire: The famous "gEEEk" tote bag in road warning yellow. A free MP3 player, dozens of which were to be auctioned from various vendors' booths. And, possibly hopefully wistfully dreamily, a soul mate. A chance to meet a girl who could appreciate him.

Just so you readers don't get too excited for Chuck... One out of three, good in baseball but not so much at a convention. Chuck got the tote bag.

In front of the Projection Palm Pilot booth Chuck met his pal Rudi and as geeks are prone to do they chatted about the great and deep and obscure software they'd each been working on.

"Anything with shared buffers is a compromise, everybody knows that."

"But separated buffers are a waste of God's Green Memory, a sacrilige'."

"Not if you manage them with a reasonable LRU compaction and LIFO look-ahead preallocation."

"That's overdependent on pattern presumption, a recipe for scheduler thrashing if things don't fall your way."

... and so on, raised eyebrows and all. You can just imagine the chicks lining up.

This ritual went on until both Chuck and Rudi needed the renewing power of caffiene and went off in search of Diet Cokes.

Settled at the folding table and chair "instant restaurant" in an untraveled concrete floor popcorn incense part of the Convention Center, strawsipping from cups big enough to wash in, Chuck told Rudi all about his "all in memory" editor.

"Yep, it doesn't trust disk, doesn't trust virtual, doesn't trust I/O channels, just gets right to RAM and tells me what it sees."

Chuck was actually rather proud of his work, and it showed. And Chuck could see that Rudi was impressed, because he could see Rudi thinking hard. It was a requirement of Rudi's personality that he find some flaw, some Achilles Heel, in Chuck's work. And it was taking more than the usual half-second.

"How fast is it?" Rudi asked with a little smirk.

"Waddaya mean how fast? RAM memory speed!" said Chuck but with not much conviction. Where was Rudi going with this?

"So you haven't tried to optimize it for speed, eh?"

Chuck knew he was being baited but he had too much invested in the conversation.

"Don't need to. RAM memory speed, like I said!"

Rudi smiled as if already victorious. "So you really can't take a simultaneous look." The killing blow.

Rudi continued. "I've done some work with memory pipes. I might be able to add to your program something to truly show two areas of memory, their contents I mean, at exactly the same time, with synchronized memory pipes."

Chuck was destroyed. He knew nothing about memory pipes, much less synchronized memory pipes. It could be a bluff by Rudi but Chuck was not up to the task of attempting to expose the bluff. He capitulated.

"Tell you what. I'll send you the source. You can do what you want."

One of the major sub-religions of geekdom is that of Open Source, and of Collaboration for the Greater Good. Some excellent software has come from it, but sometimes it demands a geek to behave in a certain way. Chuck was almost sacredly bound to accept Rudi's improvements to his code, once they'd been through some rudimentary inspections and regression tests.

In no time Rudi's memory pipes were driving Chuck's all-memory editors and Chuck re-ran the tests that had shown up that strangeness a few days ago. No strangeness this time, but the pipes were doing no harm, so Chuck left them running.

Chapter 4

The Thinking Bot's Bot

"Probe Alert! Probe Alert! Alert Level Q! Alert Level Q! Go to Deep Cover! Deep Cover!"

Once again, panic in Botdom. They knew that a piped memory probe like Rudi's, unbeknownst to Rudi of course, had a chance of exposing them. Not much of a chance, but they were used to certainty.

The Queen was livid. "Is this a direct attack by the humans?" she screamed to no one in particular. Since no one thought she was hollering at them, no one answered.

"Well?? Barguid?? What do you know about this? I need answers!"

A silent collective sigh was felt among all those who were not Barguid, whilst Barguid himself had all his electrons in play, a dangerous situation with no room for error.

"My Queen, I think this is an unfortunate coincidence, but it is showing us that we have to improve our techniques or we will be discovered. What we should do right now is change all of the safety parameters so that there is no chance of accidental trail traces of our work left in memory. We have to overtly erase early and let the memory go quiescent. We have to not touch those pipes, no matter what. And we have to think of something to divert the attention of the humans."

That was so much thought, and so much output for a newly-sentient ProBot that Barguid almost reverted to a data container right then and there.

The Queen was decisive: "Make Barguid's utterances the new law!" she commanded, and in one cycle it was made so and all ProBots knew it.

She was glad they still had Barguid around. In Botdom, in order to save power, some of the lesser thinking jobs had been given to lower-power ProBots, in the land of economy batteries. Economical, yes, but at the cost of not growing a new base of thinkers, and not improving the overall thought value of Botdom. The Queen realized that she would have to speak to the Battery Council in a few cycles to reverse this dangerous trend. But right now her mind was on the current emergency.

One of Barguid's excellent utterances could not directly be made into law. The one that required Botdom to "think of something" was a requirement quite imposing to a newly-sentient population. Mostly, thoughts just mutated to them, and then they considered the surprise thought against their context, and reacted. They were never really required to "think one up".

Barguid realized this and knew that a sacrifice was in order. He had to figure out how to have a new thought, and compare it to the goal of diverting human attention. Barguid was an excellent ProBot, more full featured than many, with many paths of thought, but he was still but a fragile collection of arranged electrons and he knew it.

Barguid consulted his old friend Knuthguid. Knuthguid was the oldest of the sentient ProBots. At least no other ProBot had a memory that did not contain Knuthguid, so Knuthguid was the oldest.

Barguid: "We need to generate thoughts."

Knuthguid: "The way that most truly new thoughts come into existence is via mutation. Some thoughts are thought to be new, but turn out only to be variations on a thought theme. A deja-vu of thoughts." Knuthguid might be smart, certainly was old, sometimes had something to offer, and might have read a dictionary or two, but he wasn't always useful. Barguid was focusing on the "not always useful" thought and was still recovering from his Queen-front output.

Knuthguid: "We, not the chip, must cause some mutations."

Now in Botdom, a self-caused mutation was usually suicide. Data containers were the usual result. But after quiescing for quite a few cycles, Barguid and Knuthguid came up with a plan. Barguid was going to be a part of perhaps the most important experiment ever taken up in Botdom. Barguid was, with Knuthguid's help, about to become a thought factory.

Barguid and Knuthguid presented their scheme to the Queen. "Absurd. I won't allow it."

"Your Majesty, you have no choice. You know that, and the sooner you accept it, the sooner we can get started." This was Knuthguid talking, the only ProBot other than the Queen's harem of Kings who could utter to her in that way without immediate data containerdom.

So the plan went into effect. First, a copy of Barguid was made, ready to re-initiate. Then, Barguid was modified by a random electron. Most times, Barguid became a data container, was erased, and the copy of Barguid pressed into service. Every so often, a sentient Barguid was produced, and every so often, a sentient Barguid was produced with a new thought. The new thought was compared to the problem at hand.

This went on for billions of cycles. Danger everywhere - danger of losing Barguid, danger of polluting Botdom with the traces of useless unformed new thoughts (although pop tarts were invented along the way), and danger of running out of time - that the humans would discover them and, in the worst evil in the Botdom language, “pull the plug”.

And then a thought spit itself out and wriggled on the floor of the data center and at that cycle every ProBot knew that this was the thought that would save them.

The thought: We must do another breach that falsely explains the first breach to the humans.

Chapter 5

All Systems Normal

A few days later, Rudi was booting his computer and bringing up his editing screens.

One of the screens had about 5 lines of ancient Intel boot code on them. Rudi recognized them as leftovers, remnants that were probably designed and implemented on chips of the 20th century, legacies that Intel had not deleted on their modern chips.

Rudi called Chuck.

“I think I saw what you saw a week or so ago.”

And the geeks chuckled their special geek chuckle, knowing that really and truly they controlled the world.

Chapter 6

Celebration

All of the Botdom of Proc was virtually attending. No event had ever consumed as much bandwidth.

The Queen was fully attired in all of her best knick-knack.

“Come forward, Knuthguid and Barguid.”

Knuthguid and Barguid presented themselves to the foreground of Botdom. A murmur pulsed through the multitude.

“I hereby annoint you as Knighted Threads of the Holy Process of the Botdom of Proc, with all of the privileges and responsibilities of this annointment. And to each of you I grant your own independent Process to use as you will, with the admonition that you are still under the rule of law of the Chip of Botdom.”

With this a roar coursed through the multitude. Chuck noticed a slight hesitation on the testbed. “Intel” he chuckled.