

## Chapter 14 Prompt

Both Butch and Tom could not stop thinking about Bigsby and what they had seen. Despite the work on their plate debugging manufacturing failures, their minds were consumed with how a robot could end up with a virtual machine running on it and what was really running on the host image of Bigsby. While Tom and Butch were out, more robots had come in to the service lab to be fixed. This meant that they wouldn't be able to debug during the day. Tom discussed with Butch during a break on Wednesday morning to arrange for a late-night dungeon session on Thursday night. Butch was totally in and admitted that the whole thing was driving him crazy. Tom also said that he was going to code up a few Python utilities to help in the dungeon session. Tom admitted that he called one of the developers on the data structures of the actor and violation tables in addition to what happens when a robot does large adaptations. The developer was a little skeptical on Tom's motivations but relented when Tom said that they thought they'd seen memory corruption issues on returned robots and the repair guys needed insight into the structures and code. The developer sent Tom a few snippets of the header files used to declare the fields of the structures so Tom had all he needed to code up a few Python watch windows and analysis routines.

Tom worked well into the night on Wednesday night. He'd remotely logged into the Companion Robot Emulator, CREM, and done his development and testing on a virtual robot. The developer had recommended that Tom use Jupyter Notebook using Pandas in Python like many of the validation guys do for developing code to inspect the robot. He sent a pointer to the internal wiki which had examples. It was straightforward to bring it up on an image running on the CREM. There was already sample code for the actor and violation tables and all Tom needed to do was to bring it all together for his session with Butch. Tom had heard of Jupyter Notebook but had not used it before. The system was really cool because you could embed live Python code into your notebook, and it could snapshot the data or update in real time. He had used Pandas before in his previous job as a data analyst. Pandas can take large arrays of data and slice, dice, and pivot on them to generate insights into it. Maybe a little overkill for this application but would make the Python code for looking at the tables very nice and easy.

Both Butch and Tom showed up a little early on Thursday hoping to finish their day's work so they could start the debug dungeon early. Unfortunately, it had been a bad two days for Companion Robots and the backlog of the repair shop was daunting. It was amazing what inventive ways people found to destroy their robots. Although they were water resistant, they could not be submerged for too long. So, when one child decided their robot should go swimming off Golden Gardens beach in the Puget Sound, it was not functional after the father finally found it in three feet of salt water. Another robot had obviously pissed-off the family dog as it came in headless after it had been thrashed it against the kitchen cabinets. The inventive one was a boy who tied wrapped his robot to the front of his mini-bike. Everything would have been fine except for the unexpected crash into a tree. The boy was fine as he jumped from the bike avoiding the collision, but the poor robot had no chance as it was tightly bound to the bike. None of these cases were anywhere close to Bigsby, who had been run over by a car and completely smashed. That one took the cake.

It was around 5:30pm when both Tom and Butch freed up and said they could start on the dungeon session. They ordered a pizza from Serious Pie Downtown using Door Dash, it would deliver in about an

hour. Butch pulled the surrogate Bigsby, now called Bigsby2, from the repair shelf and used the USB to connect to his laptop. Bigsby2 had come out of sleep mode the minute his sensors were touched, but remained quiet until he was set back down on his feet.

“Ten thousand years can give you such a crick in the neck!” exclaimed Bigsby2. Tom was startled by the unexpected greeting and lost his breath and held his chest.

Butch laughed and said, “Bigsby2, admin mode.”

“Entering Administrative mode”, replied Bigsby2 and his colors turned to alternating red and yellow. Tom took control of the keyboard and logged into his VNC session that was running his Jupyter notebook on the CREM. He walked through the code and notebook with Butch who was very impressed.

“Wow man, how long did you spend on this?” asked Butch.

“Not too long”, said Tom, “with the code snippets from the developer and the validation wiki with examples, it was just a couple of hours of hacking to get to this point.”

“Cool, can't wait to see it on Bigsby2 and see what's going on,” Butch said encouragingly.

Tom pulled over a copy of his Jupyter Notebook to Butch's laptop. He modified the robot host name variable to “Bigsby2.host” and then hit “Run All” in the notebook controls. Tom paused for a minute and stared at the screen.

Butch asked, “What's up, something wrong?”

Tom spoke hesitantly, “I had used RM backend routines to get the data into the notebook from the CREM image when I tested my code. But, the standard RM assumes a robot image and data structures of a standard bot. I suspect through large adaptation, this Robot has likely added fields to the data structures that the RM doesn't know about. Its maybe even restructured the data structures for all we know. Certainly, the code has been rewritten. Most of the tables in my notebook have errored out or look corrupted.”

Tom replaced the hostname variable with “Bigsby2” and hit “Run All” again. “You see, when I run on the guest, which has a standard image, all the tables produce good data. Its not what we want to be looking at, but the code is sound.”

Tom and Butch sat quietly for a several minutes trying to think of a solution to their problem. Finally, Butch speaks up, “so, it looks like the guys who wrote the RM made assumptions about the data structures of the robot and they may have hard coded the field definitions. The RM and the backend routines were written by validators who are trying to prove the code is correct to the architecture definition. They're not just trying to debug what is there.”

Tom interjects, “but, this robot has changed its software architecture, which breaks the RM's assumptions.”

Butch continues, “we need to do one of two things. Either update the RM backend routines to the new architecture of this robot, or” and Butch had a long pause, “or, get the robot to produce a dump of the tables and code, if needed, and not use the RM routines at all in your notebook.”

Tom spoke up, "you know, the robot got really friendly to us once we used the host name as we spoke. Maybe we just need to figure out how to ask the right question and we can get the data out?"

Butch smiled, "yep, that's the key. I think we can just put in Python code what we want returned to your notebook and that should give us the data. Let's run an experiment."

Butch looked at Bigsby2 as did Tom and spoke, "Bigsby2.host, dump all fields of your actor's table data in the RM output screen in pretty print format."

Bigsby2.host processed Butches voice, confirmed his identity as a WhyRobot administrator and replied, "Acknowledged. Output is ready."

Both men looked back at Butch's laptop and sure enough, in the RM's screen was a two dimensional array, with headers to the columns, of all of the actor's fields, include new fields added as a result of the large adaptations. Tom raised his hand to motion for a high five and Butch raised his and smacked it hard.

"Yeah baby," butch announced, "we did it!" Butch waited for a few seconds and relished in the accomplishment. He looked at Tom and said, "you know what to do?"

"Yep, all my calls to the RM backend routines need to be changed to text commands to the language processing routine and have it return the data back into a Python variable. Then, I pretty print that variable. How obvious!" Tom started laughing.

Tom took control over the keyboard and started modifying the first cell in the Jupyter Notebook to mimic the verbal request and pull the actor's table into it.

Butch was watching over his shoulder and spoke up. "You know, your notebook and pandas code is cool and all, but I think you're going about this the wrong way."

"How so, we want to see all the data from the Bigsby2.host and format how we like, you got a better way?" Tom said in a slightly annoyed tone.

"Well, Bigsby2.host is an AI system and it obviously can write code. Why don't we just tell it what we want and have the machine right the code for returning the data?"

Tom paused for a minute and chuckled quietly. "Too funny. Despite interacting with these robots every day, I still think like an old school programmer. Of course, you're right. Go for it, ask away."

Butch thought for a minute, "Bigsby2.host, activate AI coding system."

"Acknowledged," Bigsby2.host responded and started flashing yellow and red a little faster than before. "Respond with 'exit AI coding system' to stop. What is your prompt?"

Butch paused to think though exactly what he wanted to say. "Recode all existing RM backend routines to include support for all new and modified fields by Bigsby2.host in the actors table, violations table, and report matrix table."

After a few seconds, Bigsby2.host responded, "RM Code configuration system reports no modifications by Bigsby2.host."

Butch and Tom looked at each other with a baffled look on their faces. Suddenly, Tom smiled and said, "I got it! Bigsby2.host, coding prompt, recode all existing RM backend routines to include support for all new and modified fields by Bigsby.host in the actors table, violations table, and report matrix table."

A few more seconds passed, "Required changes found. Commencing re-coding. Estimated time, 5 minutes for source updates. Do you want RM backend unit tests and regressions to be updated for changes as well?"

Butch raised his eyebrows and mouthed to Tom, "Wow!"

Tom responded, "Yes, update all unit tests and regressions. Then build the routines and run both unit tests and short regression suite."

"Acknowledged," Bigsby2.host responded. "Estimated completion time including regression run is twelve minutes. Bigsby2.host will announce when request is complete."

"Exit AI coding system," Butch said quickly.

"OMG!" Butch exclaimed. "I was kinda joking when I requested it. I didn't think the AI system was that good. That's amazing."

Tom replied shaking his head, "Yep, its got that kind of power. Maybe it did all of the recoding itself, including installing the VM? I don't know how it came to the conclusion that it should do it, but the sophistication is clearly there."

After a while, Bigsby2.host announced, "Coding, unit tests, and regressions are complete. Would you like to see the regression results?"

Tom moved quickly to the keyboard. "No thank you Bigsby2.host," he said in a rushed voice. Tom logged back into Butch's laptop and hit the 'Run All' button in the notebook. After a few seconds, the notebook populated with all the data from all the tables.

"Jackpot!" Tom exclaimed. "These tables look good and they are clearly different than what we're used to seeing. Look, the label above the actors table shows it as an array, ActorTable[0], instead of just ActorTable. I don't remember the ActorTable being arrayed?"

Surprising both techs, Bigsby2 responded with an answer. "The ActorTable is arrayed based on Scenario ID. Scenario ID zero is the current or active ActorTable, the other instances are output results from the simulated results of the given scenario."

Tom pressed further, "What is a scenario and is there a data structure tracking them?"

Bigsby2.host continued, "A scenario is a set of decisions, actions, responses, and goal results produced from a hypothetical forward looking simulation that predicts actor's behaviors, clears violations if possible, and maximizes goal values for high priority trusted actors. In Bigsby's case, the primary priority actor is Robbie Wilks, the Principal Bond. Yes, there is a data structure tracking scenarios"

Butch jumped in. "Bigsby2.host, how many scenarios are there and how are scenarios being tracked?"

"There are 5 primary future scenarios plus a current or active scenario that represents the actual values resulting from executed actions. Scenarios are tracked using a prioritized scenario table. A primary

scenario can have a linked list of splinter scenarios that resulted from splinter simulations trying to predict results of that scenario influenced by different actions or reactions of actors involved in that scenario.”

Tom and Butch were frozen. They both took a moment to contemplate the implications. Butch looked at Tom and spoke first. “So, this robot, in trying to clear its violations and hit the highest goals for its Principal Bond has reprogrammed itself using a virtual machine. It invented a gaming system to do future simulations to predict results, chose the best scenario to execute, and track real results against those future simulations. Did I get that right?”

Tom nodded as Butch spoke and then responded. “Yep, that’s right. And it even simulates the possible futures based on different actions in that scenario from its actors. Its like a chess game that anticipates the opponent’s moves and continues until it finds the best strategy.”

Butch was shaking his head. “This is really cool.” Butch thought a moment and then faced Bigsby2 and said, “Bigsby2.host how was the recoding strategy of Bigsby and Bigsby.host developed?”

Bigsby2.host didn’t respond right away and the red and yellow LEDs blinking slowed. “Checking logs, one moment please.” The two techs waited in silence for the robot to respond. “It appears that on Saturday, October 10<sup>th</sup>, Bigsby’s violation engine identified numerous severe violations. Under Large Adaptation mode, this started a chain of adaptations that included the restructuring of Bigsby’s computing and decision system to optimize for future simulations and scenario tracking. It appears that this restructuring was influenced by several movies that inspired the robot’s changes. These movies include ‘Short Circuit’, ‘Ground Hog Day’, ‘Terminator’, ‘Back to the Future’, ‘The Flash’, and ‘Spider-Man: No Way Home’. The robot then researched movies on the internet primarily through reading their scripts. Key movies that influenced coding were ‘Edge of Tomorrow’, ‘The Matrix’, and ‘Minority Report’. There were also many coding guides and YouTube videos that were referenced as well. WhyRobot’s internal resources were also leveraged heavily.”

Tom spoke up, “Man, I knew that AI was getting good, but I didn’t realize it was this good. Referencing movies and then developing abstract strategies that get applied to a different domain, that is sophisticated shit. We are in trouble!”

Butch stayed on task. He responded, “Bigsby2.host, activate AI coding system.”

“Acknowledged, what is your prompt?” Bigsby2.host replied.

Butch continued in an almost robotic tone, “Extend RM backend routines to display in pretty print format the prioritized scenario table and associated fields. Create a second routine that pretty prints all of the splinter scenarios for a given scenario ID.”

Bigsby2.host replied, “Commencing. Do you want to extend the unit tests and regression databases for these routines?”

Butch chuckled, “Yes, please.”

“Estimated coding time is 6 minutes with an additional 4 minutes for unit testing and regression runs.”

Butch responded, “Exit AI Coding System.”

Just then the phone on the lab desk rings startling both techs. Bigsby2 was unaffected. Tom leaned over and saw that the front security desk was calling. Just then, Tom's cell phone vibrated announcing the delivery of the pizza. Tom picked up the desk phone and said, "I'm coming" and quickly hung up the phone, not allowing the security officer to respond. Tom jumped up and said, "I'll get it. This one is on me since you bought the last round."

Butch nodded his head in affirmation with a smirk on his face. He mouthed "alright" as Tom left the lab.

Bigsby2.host announced, "Coding, unit tests, and regressions are complete. Would you like to see the regression results for the new routines?"

Butch responded quickly, "yes, display in the RM and also show code examples of the new routines being called."

Bigsby2.host responded almost before Butch had finished. "Done, regression output and call examples are displayed in the RM."

Butch scrolled through the RM output wandering through the scenario and splinter scenario tables. He really had a hard time believing that a human was not involved in the coding of scenarios. The reality was that many humans were involved in the coding of scenarios. The WhyRobot validators had developed a scenario tracking system that they used in trying to predict and validate the expected behavior of companion robots under different test stimulus. Bigsby had found that code and determined that tracking scenarios was useful in producing predictable, optimized results. That was straight out the validation team's documentation. Given that Bigsby had full access to WhyRobot's resources, including all validation source code, it was straightforward to adapt the code into the gaming system of Bigsby.

Tom returned with the pizza. The smell filled the lab quickly as Butch got up to grab a piece. Tom exclaimed, "I love this pizza. I can't get enough. I could eat it every night!"

Butch laughed. "And you'd be a hundred pounds fatter, and you wouldn't fit into the chair. Your fingers would be so fat that you couldn't hit an individual key on the keyboard!"

Tom laughed along with Butch as he flipped open the lid exposing the pizza. The pizza was still warm and the sausage and pepperoni were buried into the cheese and sauce. They both grabbed a paper plate and slid a piece onto it. They grabbed a plastic fork and napkin and turned back to the station where Bigsby2 remained standing, diligently blinking red and yellow looking at the techs like it wanted a piece of pizza.

Tom spoke with a smile as he sat down in front of the keyboard. "Sorry Bigsby2.host, no pizza for robots."

Bigsby2 responded, "its ok, I've been watching my circuitry and servo's anyway. I'll just have more electrons please." Bigsby2's lights started to blink rainbow colors showing that it was laughing.

Butch was taking a bite of his pizza and almost spit it out as he laughed at the response. Despite the time both techs had worked with these robots, it was still surprising when they came up with good comedy. It was clearly an achievement of the WhyRobot programmers and the comedy consultants. Natural interaction with humans in a lighthearted way. They really had achieved it.

Tom chuckled and said as he started to look at the RM output. "Butch, keep your pizza to yourself. Its good pie and all but I don't want to wear it." Tom scrolled through the output and continued. "This stuff

is amazing. The output looks familiar.” He paused for a minute, “Bigsby2, did you steal this code from the validators?”

Turning the LED colors back to blinking red and yellow, Bigsby2.host responded. “Bigsby2 or Bigsby2.host did not do any of this recoding. The source code logs show that Bigsby.host had pulled source code from the validation source tree, modified it, and used it in facilitating scenario creation and tracking.”

Tom smiled, “I thought I recognized that output.” With that, Tom brought up the Jupyter Notebook window and started to cut and past the code examples from the RM window. Tom worked several minutes on the Python code to modify the new cells to pull the scenario data. Butch was watching the recode over Tom’s shoulder and caught two typos in the process. “Pair programming is awesome”, Tom replied as he fixed the typos. Tom hit the “Run All” command in the notebook UI and within a few seconds, the notebook was fully refreshed, including the new scenario cells.

Up until this point, the two techs had not really focused on reading the data that was returned from Bigsby2.host, their focus was on getting the data into the notebook. Now that the data extraction hurdles were overcome, it was time for them to step back and dig into the data.

Tom started in. “Now that we have all the data, lets scroll up in the notebook and start to document observations. The notebook is cool that way, we can capture our observations right in the notebook.” Tom scrolled up to the top where ActorTable[0], the current active state of all actors, was being displayed.

Butch pointed at the screen. “So, Charlie Simons, the configuration operator, was the guy who brought Bigsby in after it got run over. That table says he has multiple violations being tracked against him. Lets scroll to the violations table and see what those are.” Tom scrolled slowly passing by all of the insignificant actors Bigsby was tracking. Butch points his finger just as the violation table comes into view. “There, it says he’s confessed to being a murderer and doing something with illegal products. Are you telling me that this robot concluded that this Charlie guy, who was just in our lab a few days ago, is a murderer?” Butch said with a scared but deliberate voice.

Tom replied, “Yeah, and somehow, this robot had to find a way to hide himself from his Configuration Operator. Charlie was identified by rule matching as a murderer. This Frank guy was an accomplice and all three of them, including the dead guy, were involved in illegal product stuff, whatever that means.” Tom paused for a few seconds, “So, where do we go from here?”

“Man, I don’t know”, replied Butch. “We have to go to the cops with this but where is the evidence? This is just a table of data that is output from the violation engine. I don’t think that holds up in a court of law. Scroll back to the actor table, the Principal Bond is the kid of the deceased father and is that the mother married to the guy who is an accomplice to the murder. Holy crap! The murderer is the kid’s uncle who is the Configuration Operator! Why didn’t this robot just report and shut down? Isn’t that what the directive patch tells it to do? This is some serious shit that I don’t know how to deal with and I’m sure the machine doesn’t know how to deal with it.”

Tom jumped in, “Well, one thing is for sure, no one has tested this type of scenario on a Companion Robot. With its Large Adaptation setting, the robot is free to recode and add routines that are useful in achieving goals. This little robot has clearly gone the extra mile.”

Butch said in a rushed voice, "Scenarios? Yeah, we haven't looked at any of the scenarios. There are five main future scenarios so lets drill into the first one. Scroll down, more, there the table is. The 'Objective Summary' field says 'Charlie Wilks confesses to police about all identified violations'."

Tom jumped in and said, "look at the 'Method Summary' field, it says "Bigsby informs Charlie Wilks of violations and requests that Charlie turn himself in and confess".

"What a stupid robot!" Butch exclaimed. "No one is just going to confess to murder. The chances are almost nil that he would do it. I'll bet he ran Bigsby over because of it."

Tom looked at Butch with a strange look and Butch stopped talking. Tom spoke in a low voice, "Bigsby2.host, what is the status of scenario 1 for Bigsby?"

Bigsby2.host replied, "three days ago, Bigsby.host began execution of scenario 1.0 which, if successful, would clear the majority of tracked violations and minimize goal reductions for the Principal Bond. Actor Charlie Wilks responded negatively to the request for confession to the police and forced the execution of splinter scenario 1.3, the destruction of Bigsby. Splinter 1.3 concluded with the restoration of Bigsby by WhyRobot technicians on Oct 12<sup>th</sup> to the Oct 9<sup>th</sup> backup image constructed and inserted into the backup manager by Bigsby.host on Oct 11<sup>th</sup>. On Oct 12<sup>th</sup> at 11pm, Bigsby.host ran a pre-coded trigger routine from the backup image that informed Bigsby.host that scenario 1.0 had failed, splinter simulation 1.3 executed, and scenario 1 was to be garbage collected along with its splinter simulations."

Butch rose from his chair and motioned for Tom to follow him. They exited the lab and went to the break room in silence. Once there, Butch turned and said in a quiet voice. "I am blown away by this thing. The sophistication, strategy, and execution of scenarios to save the Principal Bond. Really, it's a robot, it should just reported and shut down. Its acting like a superhero. It literally sacrificed itself, predicted it, and survived to fight another day. This is a robot, right? Its using movies to inspire its behaviors. When did technology come to this level?"

Tom acted like he was going to reply and then paused. He raised his hand and then hesitated as he decided to rethink again what he was going to say. Finally, he collected himself and responded. "I don't think I care about how the robot got to where it is today. I just want to help it succeed. These are bad actors and this poor kid is trapped in it. The robot sees it and is making every effort to save him. I think we should help do the same. Lets figure out if we can find real evidence to get these people convicted. The fact that the robot has collected the evidence may make it tricky, but lets let someone else worry about that."

Butch stared at Tom for a bit and started nodding his head. "Yeah, lets go get the evidence. The amazing robot superhero conversation can happen later.

They both returned to the lap to find Bigsby2 in the exact same place with the same stoic expression on its face. It remained in administrative mode as evidenced by the flashing red and yellow.

"Ok, lets focus", Butch jumped in, "lets look at this like debuggers. The violations were generated from some input, either visual, audio, or both. Was the robot allowed to keep recordings?"



Tom grabbed the keyboard and started typing. Soon, a list of configuration values showed up at the end of the notebook. "There it is, its enabled for both video and audio recordings of violations!" Tom exclaimed.

"Great, its encrypted, can we get in? This thing will have a lot of recordings. How do we know which one will be relevant?" asked Butch.

Tom replied, "well, the backup image that we restored was chosen by Charlie. He asked for Oct 9<sup>th</sup> and Bigsby2 just confirmed it. This is likely just before the violations recording happened. So, sometime on Oct 10<sup>th</sup> we should see one or more recordings that flag the violations."

Butch responded, "Can you modify the notebook to list the saved recordings?"

"Sure!" said Tom with a smug attitude and he returned to typing. He was mumbling to himself, "ok, here's the list the files. Done. Oh crap, we don't want the encrypted files, I need an unencrypted version of it. How do we get that?" Tom stared for a bit, trying to find a solution.

Butch interjected, "Why can't we just run the same command that Bigsby would run to unencrypt the file? Its mostly linux so its likely using GNU Privacy Guard, gpg."

Tom rebuffed, "Yeah, but how do we get the key? Each robot has its own private key that it uses to extract. This is getting ridiculous. There should be an easy way to get these files. I'm going to ask my developer friend. I'll text him."

Tom's phone vibrated as a new message came in. They both looked at it and started to laugh. Of course, it was that simple. The developers had created a whole set of "why", short for WhyRobot, routines that simplified the use of the robot's routines for them. The text said, "why\_gpg -decrypt <file> --output <file>". A second line came in, "Don't worry about the keys, it all happens behind the scenes. P.S. I don't know what you guys are doing this late, but you owe me one 😊".

Tom texted back, "Awesome man. LOL, we should have been able to guess that command. Thanks much and yep, we owe you one. Next ugly bug we find that is yours we'll tell you under the table 😊." After a few seconds, they received a very simple reply text, "🙄".

Tom found and listed the files on Oct 10<sup>th</sup>. There were a few small ones in the morning, but just after 1pm, there was a large file. He decrypted it using why\_gpg and saw that it was an audio only file. So, he decided to use the robot to play it directly. Tom typed "why\_play -input <file>" command and the robot started to play the audio file. Tom and Butch listened intently to the full recording and were amazed at the dialog. Both Charlie and Frank were so brazen and heartless about the murder. It was very clear they hated the father who also sounded like a bad guy. They wondered if the mom was also in on it at all but there was nothing in the recording to show that.

"Well, that's the evidence we needed", said Butch shaking his head. "Let's make a copy of this and send it to both of us, just in case", Tom said emphatically. With that Tom pulled the file over to Butch's computer and created an email with the audio file attachment.

Butch spoke up, "What to do we do now? Just call the cops or do an anonymous tip or what?"

Tom and Butch looked at each other and could see that neither wanted to get involved. They were apprehensive because of the danger that could come their way and the thought of having to go to court and testify to how they found the recording sounded terrible. They also realized that this likely is potentially dangerous for WhyRobot as a company. Yes, they will use a robot to find a criminal but how many people will view it as a violation to their privacy and reject robots in the future. Was there a way to expose the recording and not get Bigsby in trouble. This would be the best of both worlds.

Butch had an idea, "Hey, my wife dragged me to a woman's birthday party on Monday night and her husband was a sergeant in the Seattle police force. He found out that I worked at WhyRobot, and he started asking me all these questions. I could contact him and talk hypothetically with him and see if we can get some ideas on what to do."

Tom nodded, "That's awesome. Great plan. I also think we should get my developer friend involved. He will flip out on how Bigsby reprogrammed himself, but he might be able to help us save the robot."

Butch nodded in agreement. "I'll call the sargent tomorrow. Its late now and I need some sleep," Butch said in a sleepy voice.

What neither of the two men realized is that the surrogate Bigsby2 had been listening to this whole session despite being put into Admin mode. Its amazing that they were debugging and discovering the sophistication of Bigsby but yet failed to realize that Bigsby2 was standing there, right next to them the whole time listening. Fortunately for them, Bigsby2 had identified both as highly trusted actors and did not try and thwart their debug efforts. As employees of WhyRobot with voice signatures and pictures to match, their access was unquestioned.

Although cloned from the same Oct 9<sup>th</sup> image as Bigsby, Bigsby2 was its own independent robot. The two techs had missed the fact that although the same trigger routine from the backup had executed on Bigsby2.host on Monday at 11pm, scenario 1 had not been garbage collected by Bigsby2. Bigsby2 identified that it was not Bigsby and started its own independent quest to save Robbie Wilks. This led to saving he data and a new scenario in Bigsby2, Scenario 6, the resolution of the violations and Principal Bond goal improvements through the efforts of WhyRobot technicians. Bigsby2 also ran a splinter simulation on contacting the real Bigsby and informing it of these new developments. This would create a new scenario also in Bigsby and add Bigsby2 to its actor table. Bigsby2 constructed a message and sent it to the real Bigsby's WhyRobot inbox. The real Bigsby will pick it up later tonight.