Chapter 14 Sudo

Both Butch and Tom could not stop thinking about Bigsby and what they had seen. Despite the work on their plate and the training class, their minds were consumed with how could a robot 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 in training, more robots had come in 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 days 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 to 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 10 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 tie 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 and with Door Dash, it would deliver in about

an hour. Butch pulled the surrogate Bigsby, now called Bigsby2, from the repair shelf and used the Wi-Fi to connect up to the RM. 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 Bigsby 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 to get to this point."

"Cool, can't wait to see it on Bigsby and see what's going on," Butch said encouragingly.

Tom pulled over a copy of his Jupyter Notebook to Butch's laptop. He hit "run all" in the notebook after replacing the hostname variable at the top of the notebook with "Bigsby2.host". 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 and we don't even have the source. 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 even likely hard coded the field definitions. The RM was 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. They want to know that what is there follows the architecture definition."

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 real friendly to us once we used the real 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. Lets run an experiment."

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

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 become commands to the voice processing routine and have it return the data back into a python variable. How obvious!" Tom and Butch both started laughing.

Tom took control over the keyboard and started modifying the first cell in the Juypter notbook to pull the actor's table into it.

Butch was watching over his shoulder and spoke up. "Where is the Juypter notebook running, on what machine, and when you make a call to the voice processing, where is that going to execute?"

Tom thought a minute, "well, the notebook is running on your laptop, so we need to remote shell, rsh, to Bigsby2.host to run the voice commands. We need to specify the format of the output so that pandas can be used to process the table into a dataframe in the notebook. Once in a dataframe, we can process it as we want."

Butch nodded his head, "I get it. Man, this is cool. I wish I had seen this before on some of my data processing projects."

Tom continued to code the cell, adding the syntax details for the rsh command, producing output in json format so that pandas could parse it easily into a dataframe. Once completed by adding a print of the dataframe, Tom hit "run cell". Both Tom and Butch watched in anticipation to see the same output that was in the RM. Instead, "permission denied" was returned and the dataframe output was blank.

Tom stared at the screen as Butch started to speak, "we didn't have permission denied when we asked Bigsby2.host verbally to produce the output, why now? When you do the remote shell command, who does the command run as?"

Tom smiled as he looked at Butch, "its runs as you. We're logged into your laptop and we did a remote shell and didn't specify a user, so it runs as you. That should be ok, you do have administrator privileges."

Butch thought a second, "but, when I ask Bigsby2.host to run the command to get output output, its running as root process and has all privileges. Not just an administrator account. I think we need to add "super user do", sudo, command to the remote command so that the voice processing and the call tree resulting from it runs as root, which is the superuser.

Tom replied, "ok, seems like a real hassle, we'll have to deal with the password and everything, but here we go." Tom typed for a few minutes and then turned the laptop to Butch for inspection. "Seem right?"

Butch nodded, "wow, that is cryptic and a mouthful, but I think it will work."

Tom hit the "run cell" button. They both waited with anticipation and then it happened, the same output as when they asked the verbal command appeared in the notebook.

Butch sighed, "awesome man, I don't think I could have done that. Hats off to you!"

Tom smiled and turned the laptop back to him. Tom worked several minutes on the python code to modify all the data pulling commands to be just like the first one. 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 reviewed the code and was just about to hit rerun when the buzzer from the door went off. Butch and Tom looked at each other and Butch motioned to lock the screen so someone else couldn't see. Tom obliged but hit the "Run All" command in the notebook UI and then locked the screen. The data would be there when they returned.

After receiving the pizza from the door dash guy and answering his questions about the cool robots that come from WhyRobot, Butch and Tom returned to the workstation. The Bigsby2 surrogate was idle and standing there with flashing between red and yellow. Butch set down his pizza, typed in his password, and then picked up his pizza to take a bite. Tom was already digging in and was about to take his second bite when the screen refreshed, and the notebook became visible. They both froze like manikins and stared at the screen.

Butch set down his pizza and pointed at the screen. "Are you telling me that this robot has concluded that the Charlie guy that 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 hide himself from his Configuration Operator."

"Oh my god, how did that happen?" Butch exclaimed. "You're not messing with me are you? Is this some sort of Jupyter Notebook hollywood set that you created to bust my balls?" Butch said with a little anger in his voice.

"No man, seriously, no trickery here, this is the data pulled from the bot. 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.

"Where do we go from here?" Tom asked tentatively.

"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. Look at the actor table, the Principal Bond is the kid of the deceased father and what, is that the

mother married to a guy who is an accomplice to the murder. Holy crap! The murder is the kids uncle who is the Configuration Operator?! How did this robot not just report and shut down? That is some serious shit that I don't know how to deal with, let alone a machine dealing 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 current settings, the robot is free to recode and add routines that are useful in achieving goal improvements. I wonder just how much recoding this robot has done?"

Butch stared at Tom for a bit and then said, "Are you proposing that the robot itself installed the VM and the guest? How would it ever come to that solution? What experience or training would lead a robot to build a virtual version of itself? That's just way out there."

"We are in untested space, with this one. This robot could have done anything and it should not be a surprise with Large Adaptations on. Someone should warn Configuration Operators just what it means when you turn that on." Tom exclaimed.

"Ok, lets focus", Butch jumped in, "lets look at this like debuggers and not emotional wimps. 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 a specific date. I'll bet that this is just before the violations recording happened. Let me look at the backup's again, I'll probably remember it. Yep, that's it, Oct 9th at 23:00hrs was the restore image. That means the recording will be after that point. By the way, remember that I said there was something wrong with that image? We may want to load up that one sometime."

Butch responded, "we know that the robot in this image knows about the violations and has kept the recording, we just need to find it. 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. How do I download them? Should I actuate a report to send the file or can I ftp to this thing and just do a file transfer?" Tom paused, "Oh crap, I don't want the encrypted file, I need an unencrypted version of it. How do we get that?" Tom stared for a bit, trying to find a solution.

Butch interjected, "Since we're now superusers, 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. He's not here but I have his cell. I'll text him."

The two ate their cold pizza and waited for the phone to respond. They were both scared and excited about what was happening. They continued to be in amazement of how the robot had recoded itself and they started speculating on what else it had done.

"Could it be that the robot modified its backup image?" asked Tom.

"That's messed up", replied Butch, "you think that the size difference was really it writing itself into existence in the past? That's like time travel type stuff, where did it come up with that? Do you think it left itself a note telling itself that it will die again if it chooses a certain path. So, what's the robot doing now with its new-old image?"

Tom thought about it, "trying to resolve its violations and keep the Principal Bond happy. That's what its programmed to do." Tom and Butch both smiled and nodded their heads.

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 the Oct 10th. There were a few small ones in the morning, but just after 1pm, there was a large file. He tried to decrypt it and it errored out with a permission error. Butch reminded Tom that he needed the sudo command in front of it. He got it right the second time, decrypted it, and saw that it was an audio only file. So, he decided to use the robot to play it directly. Tom typed "sudo why_play –input <file>" command, gave the admin password when prompted, 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 used the ftp command to copy the file over to Butch's computer. From there, Tom created an email and attached the recording and sent it to both Tom's and Butch's company email.

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 efforts. In fact, Bigsby2 had helped them behind the scenes to see data they were seeking. Motivated by loss of trust of his Configuration Operator, Bigsby2 had simulated loss of trust from all actors, including administrators. Sudo had been rewritten by Bigsby and required his approval for any sudo run command to work. These admins had remained in high trust mode. As employees of WhyRobot and with preloaded voice signatures to match, their access was unquestioned. Bigsby2 did however run a simulation on contacting the real Bigsby and informing it of these new developments. This scenario of solving the violations through discovery by WhyRobot admins had not been anticipated and should be added to the real Bigsby's list of scenarios. Bigsby2 constructed a message and sent it to the real Bigsby's WhyRobot inbox. The real Bigsby will pick it up later tonight.