

Chapter 69 - Cyberspace Foray II

Surprisingly enough, Kill Joy's bombshell about my Avatar getting shredded in Cyberspace leading to my actual death didn't really faze me.

It was pretty much what I'd been expecting, given the circumstances.

While there wasn't any HP to abstract the damage in this world, dying in the *game's* Cyberspace still meant hitting reload—essentially death, if you really thought about it.

'*Note to self: Don't die,*' I mused as I processed the mechanics Kill Joy's golden avatar had just laid out.

"Got it," I said out loud before getting back on track.

The floor tiles had lost all their appeal after that brief interruption, so I kept moving, eager to find something more interesting.

My feet carried me past a bunch of different server sections—buildings or even whole blocks sectioned off from the main Cyberspace. They all screamed, "Keep out!" with fences, walls, and even some rare, glowing force fields around the entire property.

On a whim, I decided to check out one of these rare force field-protected spots.

It was just too "sci-fi" to resist—who wouldn't want to know what touching a force field felt like, after all?

I approached slowly and carefully, half-expecting Kill Joy to pop up and warn me that the force field would disintegrate me or something. But as nothing of the sort happened, I felt more confident in my plan.

Stretching my arm out as far as I could, still a bit wary of any potential nasty surprises, I finally managed to brush the tip of my fingers against the force field.

The feeling was beyond strange for multiple reasons.

First off, the physical sensation was weird on its own. My fingers sank slightly into the force field before they got "stuck" and couldn't go any further. I could pull them back just as easily as they'd gone in, but trying to push through felt impossible, like there was a solid rock-crete wall right behind that initial layer.

Secondly, and more immediately worrying, was the strange shiver that ran through my body the instant my fingers touched the field. It wasn't a cold shiver, like something bad was about to happen, but a very specific kind of shiver—I instinctively *knew* the field had just scanned me. I had no idea to what level of detail or what info it had grabbed, but I knew for *sure* it had scanned my Avatar completely.

"Hmm..." I mused aloud, hoping Kill Joy might jump in with some explanations if I threw out some thoughts. "So the force field scanned me; I guess it now has my signature or

something. A netrunner on the other side could potentially figure out who I am using that signature. Kind of like a fingerprint?"

Glancing over at the golden Kill Joy, I saw him nodding, signaling me to continue. "The field itself seems to be made of multiple layers. The outer one is super easy to get through—my fingers just slide right in—but behind it is a solid wall. No matter how hard I try, I can't push through. It's like trying to break into a solid rock-crete house with just my fingers," I continued, reflecting on the sensations I felt while interacting with the force field.

As I pondered about this, I suddenly had a small epiphany, recognizing the specific timing of the scan. "Ah...! So the first layer is some sort of scanning field. It purposely lets you in easily to do that weird data-handshake thing, just like with the tiles!"

Kill Joy nodded, looking pleased with my deductions. The fact that the first layer was there to facilitate a data-handshake but also had an added trap for a full-scan got me thinking, "I wonder..."

I conjured up my cerebral interface and quickly searched for Cyberspace handshakes. Luckily, the whole system had a search bar; otherwise, I'd probably spend hours digging through all the apps and options. Instantly, a list of dates, times, and addresses popped up, the newest one matching the scan from a few seconds ago.

Curious to see what such a handshake looked like in code, I opened that entry and was immediately hit with lines and lines of information.

"Whoa..." I couldn't help but exclaim, earning a chuckle from Kill Joy's alter-ego.

The data-handshake was way more detailed than I had expected.

It didn't just show the signature and time of day but also the type of handshake—listed as "non-forceful probe"—the exact location on the force field, and dozens of other lines that were crossed out as irrelevant. There were fields for "detected injection," a binary for "breach attempt," and loads of other netrunner-related data.

"Does every handshake share this much stuff...?" I hesitantly asked Kill Joy, slightly worried that my dreams of becoming a fledgeling netrunner for the Operator meeting would die before I could even really get it off the ground.

If every handshake gave away this entire suit of information, there was no way I could stay below the radar of anyone without some serious cloaking—another one of the many things I hadn't learned a single thing about so far.

"I could give you all the answers, but I really don't feel like it, girl," Kill Joy's alter-ego replied, sounding a bit petulant.

'Wow, okay. Not like I didn't pay you to do exactly that or anything; stupid program,' I thought, a bit taken aback by his flat-out refusal. *'What good is a teacher that doesn't want to teach?'*

It didn't take long for me to realize why he'd refused, however.

I had to admit, I'd been a bit too narrow-minded with my request.

I pulled up the data-handshake from one of the tiles I'd recently passed over and got my answer immediately—the likely reason Kill Joy had clammed up was because I could find the answer myself.

The tile's handshake only listed the exact info I'd expected: Date, Time, Location, and Signature.

That brought up another obvious question: Why were the two handshakes different?

I mulled this over for a few minutes, getting up close and personal with both the tiles under my feet and the force field a few centimetres away. But it was no use. Staring at these constructs in 3D space was no different from staring at the floor or a wall.

There was no way I'd get any answers like this.

Resorting to my second-best option, I asked Kill Joy, hoping he'd be more willing to spill the beans this time.

"So, why are the two handshakes different?" I asked, figuring this was something I couldn't just learn on my own.

"Because they're *different* handshakes," Kill Joy answered with a smug smile, leaving me hanging before continuing. "There are three types: Basic, Advanced, and Full."

He pointed at the tiles below our feet. "These use the Basic handshake. They only need your digital signature to function, so a Basic handshake is enough. Force fields like that one," he gestured to the radiant wall of light next to us, "use the Advanced handshake. They gather more data to figure out who might be trying to access their info. It helps the owners of the server track you down if you cause any undue trouble on their address."

Then he pointed to the massive, towering data-structures in the distance—the megalithic, unmistakably corpo-buildings that dwarfed everything around them. "And those ones would use the Full Handshake. That one gets essentially every piece of information transferred that you could possibly imagine wanting in Cyberspace. Things such as connection type—as in, how you are connected to Cyberspace—your equipment, your cybernetic implants that are active at the given moment, and so on. It's a very intrusive, all-encompassing sort of handshake, only reserved for the most secure of areas."

My eyebrows shot up involuntarily.

The sheer amount of privacy violations implied was staggering, but then again, this wasn't my old world. Privacy laws didn't exist in Neon Dragons—if a Corp wanted your data, they'd just take it. After all, who was there to stop them? The police that were already owned by them? The politicians that were literally the CEOs of the corporations themselves?

"I assume there are ways around that, right?" I asked, hoping Kill Joy might delve into cloaking or masking; basically [Stealth] for netrunners.

“There are,” he nodded sagely.

I waited for about half a minute before letting out a big sigh.

As far as I knew, cloaking and masking weren’t covered in the SPG-01 shard.

It was netrunner-specific stuff used for illicit activities and had little use outside that, so it wasn’t included in the “consumer” grade SPG-01 tutorials aimed at low-level corpo netrunners. You didn’t want to teach your low-level employees how to mess with the corporation’s servers undetected, after all.

Deciding I’d wasted enough time just staring at the force field, I fired off another question to Kill Joy, hoping this one would be straightforward. “How do I get inside if this force field exists to keep me out? I’d have to breach it somehow, right?”

He nodded, encouraging me to continue.

“If I want to breach it, I’d need some kind of program or quick-hack to get through. What’s the difference between the two? Quick-hacks are faster, right? So why would I ever use a program then?”

Out of nowhere, a chair appeared beside me. I got the hint and sat down, expecting this explanation to take a while.

As soon as I was seated, the golden man started speaking.

“First off: Getting through something like this force field is easy. They’re mainly designed to signal that whatever’s on the server behind it is off-limits to the general public. They aren’t *meant* to keep people out, just to inform them they shouldn’t enter uninvited,” Kill Joy explained.

He snapped his fingers, and the force field shattered in a 3x3 metre square, like a giant gate opening at his command.

I had to admit that I was impressed by that, having just tried to push and pull at the invisible rock-crete wall behind the initial layer.

But I knew I’d been going about it the wrong way as well. Physical force didn’t really work like that in Cyberspace, nor would my pushing and pulling have done anything to a real rock-crete wall either; so I didn’t really even try anything that could have possibly worked to begin with.

“Much like your personal hardware is limited by heat, capacity, and RAM, so are the servers that host all the data; or rather, the servers used to protect the servers that host all the data,” he continued, conjuring up the blackboard from earlier.

He drew a large rectangle and a smaller one next to it, pointing at the large one. “This is the data server. It houses the information the corporation, business, or person wants to keep. It primarily consists of storage capacity, maybe with a few basic data-aggregation routines running. They’re incredibly easy to access and navigate, making data-theft a real breeze.

This is mainly because building a large data server with enough capabilities to keep a netrunner out is extremely expensive.”

Kill Joy made sure I was paying attention before pointing at the smaller square on the blackboard. “This is the security server. It’s basically a deck, just larger and more powerful than the personalised ones netrunners use. It comes with the same heat, capacity, and RAM stats as your deck, just with a whole lot more of it.”

He drew a bubble around the large rectangle as he continued, “The security server acts as a defensive layer for the data server. It hosts all the defensive protocols, daemons, walls, and whatnot to keep netrunners from accessing the data server. This is where the *real* magic happens in netrunning, both for attackers and defenders. The defending netrunner usually has the home advantage, with existing defences and daemons, and they can offload some of their heat and RAM onto the server. The aggressor, meanwhile, has to get around these pre-programmed defences and the defending netrunner, all with their own deck’s capabilities.”

My eyes widened as I realised just how daunting the task of being an aggressor was. I’d always known that invading a defended system was tough, with a higher difficulty rating in the game, but I’d never really thought about the “why” or the “how” in depth before.

Curiosity got the better of me, and I asked, “So... how does anyone even succeed at data-theft then? If the defending netrunner has multiple times the power of the attacker and pre-installed defences, how could anyone ever even *think* attacking a protected server was possible?”

With a big grin, Kill Joy replied, “It’s actually quite simple if you think about it: You either power through before the defending netrunner realises what’s happening, do it sneakily enough that they never knew you were there, or you just attack the netrunner directly. Nothing says you have to fight the server itself, after all.”

His words made a lot of sense in hindsight—I had been so focused on the advantages of defending that I hadn’t considered any of the downsides at all.

The defending netrunner had to realise their server was under attack before any of those advantages would kick in to begin with. If you were done with your heist before the defender noticed, all the server power in the world wouldn’t help them turn back time and lock up the data again.

When Kill Joy mentioned attacking the enemy netrunner directly, I couldn’t help but think back to the game’s mechanics for netrunner combat either. They were pretty involved—more so than other games of its kind—but I had a feeling the in-game representation was quite different from the real deal; something I was low-key kind of hoping we would get to in one of these lessons, but the shard’s rundown hadn’t listed it explicitly, which made me doubt we’d actually get to it.

It was likely something I’d have to teach myself how to do once I entered Cyberspace proper... Definitely a daunting task, but one that would invariably be necessary if I wanted to sell myself as a netrunner.

“Now to answer your actual question,” Kill Joy’s words snapped me out of my thoughts about netrunner combat. “When it comes to deciding whether to use a program or a quick-hack, the answer is simple: Do you need to get in *today*, or do you need to get in *someday*? If it’s *today*, you rely on quick-hacks. If it’s *someday*, then a program is better.

“Programs are purpose-built and can run autonomously, while quick-hacks need the injector to be present—whether that’s you, a drone, or something else that can inject quick-hacks. If you’re planning a heist ahead of time, getting access to a nearby upload connection and using a purpose-built program would be ideal. But that very rarely happens. So in 99% of cases, you’ll be relying on quick-hacks for your offence. If you’re defending, things are a lot different, but we’re not really talking about that yet. As for which quick-hacks to use on offence, you should already know the answer from our first lesson.”

He looked at me expectantly, clearly turning this into one of his impromptu tests.

Luckily, this one was easy.

“Subroutines whenever possible, Segments if I have no other choice or encounter something unexpected,” I answered easily, remembering the first lesson about quick-hacks like it was yesterday.

Back then, he had only shown me examples with doors, but when scaled up, how different was a door from a security server *really*, if you thought about it? They were both designed to allow things to pass-through or deny entry, one was just a tad bit more sophisticated than the other.

Abruptly, Kill Joy tossed two shards at me, which I managed to catch without much trouble thanks to [Perform], probably the [Juggling] portion of it coming in handy.

I inspected them curiously, unable to tell them apart at all. They looked exactly like the SPG-01 shard, just without the fancy design—simple, sleek, and monochrome.

I glanced up at Kill Joy, waiting for an explanation.

“You said you wanted in,” he said, gesturing towards the force field. “Those are blips—shards with pre-installed quick-hacks that self-destruct after a few uses, three in this case. Try them out, get a feel for how quick-hacks work, and explore the world behind the force field.”

Excitement surged as I quickly slotted the first blip, only to be scared half to death as a giant image of Kill Joy’s golden, laughing face appeared in my cerebral interface. “Whaaa—!”

As fast as it appeared, it disintegrated, and the blip refused to load. Taking it out, I looked over at Kill Joy, my heart still racing, and groaned at his smug expression.

“You should *never* slot something given to you by someone else without checking it for contaminants first. Never assume that even people you trust implicitly won’t give you infected shards,” Kill Joy lectured me in his typical lecturing tone.

I couldn’t argue because he was right.

Slotting something into the neck-slot gave it implicit access to your cerebral link—the very connection between your brain, body, and cybernetics. If the shard had been harmful, it could have done serious damage, especially since the ICE—Intrusion Countermeasures Electronics—on my cerebral link was probably far from being top-of-the-line.

The best defence against invaders was not letting them access anything that needed protection in the first place. That's why layers were crucial in cybersecurity. The more layers between the data and the attacker, the harder it would be to access—in theory, at least.

There was one thing I could be mad at him about, though. “You never even taught me how to check for something! How was I supposed to even check them if you never taught me *how?!?*”

Despite my very reasonable complaint, Kill Joy kept his smug expression and said, “You didn't need to know how; simply stating that you didn't trust them or wanted to check them would have been a pass, girl. Not everything needs to be possible for you to be tested on.”

I let out a long groan.

Kill Joy's constant “better than thou” attitude was really starting to grate on me. I had initially found it endearing and didn't mind it much during our earlier sessions, but spending so many consecutive hours with him as the Operator deadline drew nearer was pushing my patience.

‘Keep your calm, Sera. It's literally just a program. The second you slot out the shard, it's gone. So you can “kill” him whenever you want; no need to get upset.’ I reminded myself mentally.

Then, holding up the second blip, I said demonstratively, “I don't trust this shard; I'd like to check it, but don't know how.”

“Oh, why, oh my, would I ever have forgotten about this!” Kill Joy's alter ego played up the melodramatic realisation as he often did in these situations. “It's a good thing you remembered not to slot in things that you aren't sure are safe! That could have been a *really bad* problem if you did!”

‘Breathe, Sera. Just breathe.’

“To check a shard, it's quite simple in most cases: Create a sub-section in your own system to load the shard into. If there are any harmful parts that trigger on insertion, they'll harmlessly wreak havoc in that little extra part—allowing you to delete them along with the partition. If there are harmful aspects that trigger later, you'll need more sophisticated analysis, but that's beyond today's lesson. We'll cover those later,” Kill Joy explained, drawing something on the blackboard.

As I watched him, I realised he was drawing up instructions for my cerebral interface.

I figured he wouldn't lead me astray again; Kill Joy wasn't one to waste time repeating lessons. Following his guidance, I quickly found the partitioning part and created a small section to load the blips into before using them.

Slotting in the second blip, I was immediately bombarded with a slew of warnings.

My cerebral interface flashed signals in the corners of my vision, alerting me to something. Opening up the interface, I found a message waiting for me:

[Partition 1 overloaded due to inserted subroutine. Enlarge partition? Y/N]

It took me a moment to understand what it was asking, but as realisation dawned, I simply denied the request, only to be greeted by another message:

[Partition 1 overloaded due to inserted subroutine. Partition enlargement denied. Subroutine will be stopped to maintain partition limit and not impact overall performance.]

As soon as the message appeared, I saw, on the second “screen” inside my head, the laughing face of Kill Joy freeze in place.

“I got the contaminant frozen. Now what?” I asked Kill Joy, unsure of the next step.

Deleting the partition would get rid of the contaminant, but that wouldn’t help me use the blip—the blip was loaded in the same partition, after all.

Kill Joy smirked and replied, “Good job freezing it. Now, simply create another partition and transfer the blip’s data there, minus the frozen contaminant. Think of it like moving your clean laundry away from the dirty stuff. That way, you can still use the blip without the nastiness.”

Following his advice, I created a new partition and carefully transferred the blip data, leaving the frozen contaminant behind.

It was surprisingly easy; my cerebral link did most of the work.

I just had to think about what parts I wanted to move, and the software inside my head took care of the rest. Once I was sure everything had been moved over, I deleted the first partition.

I watched the frozen, laughing face of Kill Joy melt into pixels before disappearing completely, along with the second “screen” in my head.

*‘Well... That was incredibly satisfying. And not **just** because it was his face being melted apart,’* I thought, feeling a bit smug myself.

“Judging by your expression, it seems everything worked well! Good job, girl,” Kill Joy commented, giving me an encouraging thumbs-up.

He wasn’t so bad once he stopped being all smug.

“Now, carefully read through the blip’s description—but remember, this is just the description I entered. Whatever a blip’s description says doesn’t *necessarily* reflect what’s actually in the data. That’s the downside of blips; you can’t delve into the code to see *exactly* what it does. We’ll cover how to get a rough idea of a blip’s functionality later on in the shard’s sessions.

For now, just be aware that descriptions aren't always accurate," Kill Joy warned me, gesturing for me to return to my cerebral interface.

I pulled up the blip's description:

[Description: Subroutine to enter any protected space entirely undetected, created by the majestic, beautiful, and erudite Kill Joy himself. Guaranteed to have zero side effects or weaknesses. — Uses Remaining: 3/3]

I rolled my eyes at the over-the-top description but couldn't deny that I had little doubt about its authenticity. Kill Joy was, after all, a [Transcendent]-ranked individual in the world of netrunners, programming, and quick-hacks.

If anyone could create a Skeleton-Key blip, it would be him.

"Is the last part, 'Uses Remaining 3/3,' part of the entered description, or is this more objective information? Where does it come from?" I asked the golden avatar.

This seemed like pretty important intel to know.

If anyone could just enter whatever they wanted, you could easily get scammed on the number of uses if you didn't know how to check the real number.

Kill Joy nodded in satisfaction at my question. "Very good. I was hoping to hear a question like that, girl! The answer is simple: It's an objective guess based on your software's understanding. So you can generally trust that the number is *mostly* right. There are very few instances where it will be wrong, but they do exist. If you run into a type of quick-hack you've never seen before, it might guesstimate the number of uses wrongly. But for the most part, it will be accurate 99% of the time."

I breathed a quiet sigh of relief, glad that this was one thing I didn't have to worry about too much in my netrunner career.

I quickly slotted in the first shard again, this time with proper partitions in place to catch any nasties from Kill Joy.

After cleaning the blip and having it ready to use, I also checked its description, figuring that it was likely different from the other blip:

[Description: A collection of quick-hack segments designed to allow undetected entry into any protected space, created by the majestic, beautiful, and erudite Kill Joy himself. Guaranteed to have zero side effects or weaknesses. — Uses Remaining: 3/3]

'Hmm... This would be pretty useless in the real world as is,' I thought.

Quick-hack segments were valuable for their flexibility and utility, especially since they didn't need specific targets spelled out. If the blip just included a pre-arranged set of segments, it wouldn't be any different from a subroutine, except slower since segments are always slower than subroutines.

“What segments are in here? Can I arrange them however I like? If not, what’s the point?” I asked Kill Joy, pointing at the shard in my neck.

Once again, he gave me a satisfied smile, happy with my question.

“Good, good! You’re learning fast, girl. Very good. Now, inside the segment blip, you have another option besides just pulling up the “Description”. You can also ask for “Contents”, which will display *exactly* what segments there are, with a caveat: You can only read their *descriptors*, not their code. After all, blips aren’t meant to give you free access to copy your own library.”

Nodding in understanding, and seeing the point of how it was set up, I called up the blip’s Contents screen.

[==Contents==]

[Segment Containers: 1/1/1, 1/1/2, 1/2/2]

[Subjects: Layer, Wall, Gate, Door, Compartment]

[Verbs: Open, Propagate, Search, Unlock]

[Adjectives: Forceful, Fast, Quiet, Reckless, Careful, Skittish]

My eyes widened at the selection, my mind immediately racing with potential use cases.

This blip would undoubtedly cost an absolute fortune in the real world, considering how many segment pieces were contained within—not to mention the unspoken quality of each one, having been created by Kill Joy himself.

‘I wonder if I could somehow copy the blip’s data...?’ I couldn’t help but wonder, but quickly dismissed that thought. If it were possible, every half-baked netrunner would have already done it.

The SPG-01 shard wasn’t exactly a super-secret item, after all.

Now, more excited than ever to try things out, I stepped up to the force field and concentrated on the blip inside my head, assembling my very first segment quick-hack using Kill Joy’s blip in order to break in...