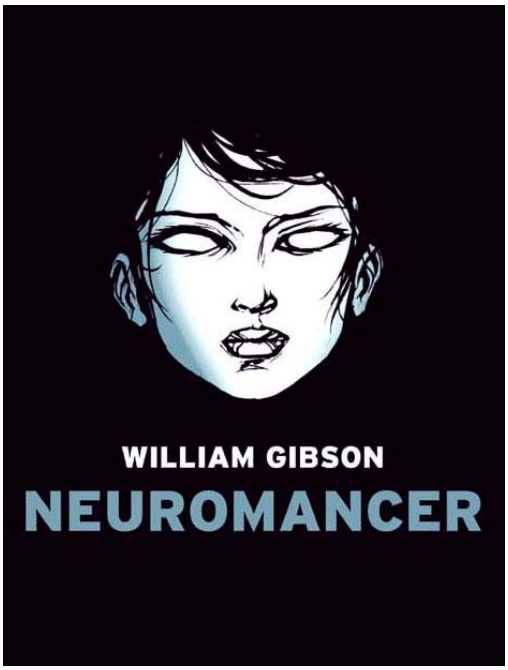
GM Don't List #4: Thou Shalt Not Hack

by Justin Alexander
October 3rd, 2017



A cyberpunk character concept I would dearly love to play some day is that of the uber-competent hacker: Case from *Neuromancer*. Batman's Oracle. Edward from *Cowboy Bebop*. Boris Grishenko from *GoldenEye*. Luther Stickell from the *Mission Impossible* movies. Half the main characters from *Ghost in the Shell*.

In my ideal version of the character, I'm the guy who stays in the van three out of five times, providing overwatch and support for my teammates while they mount their raid.

So why haven't I played this character?

Because what I've discovered is that a surprisingly vast number of GMs seem to consider the entire concept of using hacking to solve a problem to be some sort of anathema. So even when I've *tried* to play the character concept, I've ended up not actually being able to play the concept.

- Hack the security cameras to scope out the interior of the building you're raiding? Can't do it.
- Hack the security guard's cellphone to track her movement? Impossible!
- Play R2-D2 in a Star Wars game and hack an electronic lock? No way. Pull out your lockpicks!
- Hack the rigging ports on the pursuit car to seize control of it? Obviously no one would *want* you to remotely seize control of a vehicle, so they would build a perfect security system that was completely unhackable, and therefore you can't hack it.

(That last rationalization seems to crop up a lot. It's like saying that obviously no one would *want* you to poke them with a sword; ergo it's impossible to hit someone who's wearing full plate.)

Quibble here and there with the plausibility of some of these scenarios in particular settings, but I've seen this behavior even in settings and games which include mechanics for handling these specific types of hacks! What I'm talking about is a systemic pattern of behavior in which the hacker basically can't do their thing. It's the equivalent of finding an *antimagic field* everywhere you go in a D&D game, except that I've found it to be a peculiarly ubiquitous attitude.

Of course, flat out denial isn't the only way this manifests: Setting disproportionately high difficulty numbers or using roll to failure techniques are probably the most common versions, actually.

I've found this particularly pernicious in many convention scenarios: The designers of the game want to show off its breadth, so they include a hacker archetype pregen. But the volunteer GM running the scenario subscribes to the doctrine of Thou Shalt Not Hack, so the pregen is a trap and the person picking it finds themselves sidelined for four hours.

The worst case scenario is, sadly, one of my favorite games: *Eclipse Phase*. Wanting to show off everything that's possible in this cool, kitchen sink transhuman setting, the designers regularly include an infomorph pregen: A character without a body who exists only as a digital construct and can only take actions through the Mesh network.

Combine that with a GM who doesn't allow any meaningful action to take place through the Mesh network (which I've seen happen either first- or second-hand in no less than four convention scenarios) and you have a character who literally can't do *anything*.

Many of these GMs don't seem to be consciously aware of what they're doing, so you'll even find them saying things during character creation like, "Who wants to be hacker?" I used to hear that and think, "Okay! This guy is going to actually let me hack!" But, oddly, no. They recognize on a conscious level that a team of cyberpunk characters is supposed to have a hacker, but when it comes to actual play the hacker nevertheless finds themselves stymied at every turn.

EMBRACING THE HACK

I suspect part of the problem here is that a lot of GMs reflexively cling to the modes of play they learned running D&D dungeon crawls. Their expectation for how a facility raid is supposed to play out features people physically sneaking around and getting ambushed by security guards, and the hacker's attempt to grab the security cameras disrupts that expectation. Their vision of the game world (inaccurately) doesn't include hacking, so the hacker's solution to any given problem comes out of left field, and the GM reflexively shuts it down.

This is, obviously, a form of railroading: A preconceived idea of not just how a specific problem is meant to be solved, but a broad preconception of how entire *classes* of problems are supposed to be solved.

So the solution to this problem is relatively simple: Don't do that.

Conversely, however, hacking shouldn't be a magic button that can trivially solve all problems. When that happens, it creates a spotlight problem where the hacker upstages every other character and flattens the challenges presented by the scenario.

To counteract this problem, there are a couple things the GM should do. First, check the potential consequences a hacker faces: They should be comparable to those faced by other types of action. (Just as the hacker should not find it impossible to hack an automated car; the hacker themselves should not benefit from a foolproof firewall.) Second, check your vectors: Make sure that "solving" the scenario requires a multi-step resolution and, importantly, make sure that hacking can't be used to trivially solve all the vectors.

The most obvious example of this is, "I can't hack that system until you plug in my remote router!" But it can become an easy trap to always design scenarios in which the team does a bunch of stuff and "unlocks" the hacker so that the hacker can then win the day. Look at ways in which hacks are invaluable at the beginning and in the middle of scenarios.

Also remember that you don't always need to lock these things in: Players hot-swapping in vectors you'd never thought of to solve their problems is what makes the game fun. Generally speaking, the rest of the group will find ways to advocate for plans which feature the strengths of their own characters if you give them the chance. You can encourage that by creating scenarios which require multiple problems to be resolved simultaneously. Also experiment with using hard scene framing techniques to move the action "onsite", which will discourage the players from lingering in remote "planning" sequences where the hacker (and only the hacker) is capable of taking direction action.