Scalability, Fidelity, and Containment in the Potemkin Virtual HoneyFarm

I found a lot to like in this paper. The topic is very close to my own research interests, as I am coming to understand them. So, I know a little bit about the topic already, and I'm interested in learning more. There's also some new (for me) ideas here for me to add to my growing understanding of the topic. Finally, I feel as if there some areas in which I might be able to contribute in this area of research, as my experience grows.

The big ideas in the background section are fairly familiar, how to dedicate enough resources to the task of providing a believable high-interaction system while containing any damage that might be done as a result of allowing infection/exploitation. I was a little surprised at the discussion of legal liability, particularly since literally millions of bots located in homes, businesses, schools and colleges around the world are clogging the Internet and spreading infections on a biblical scale. On the other hand, as Eldridge Cleaver instructed us, there are many areas in which not being a part of the solution makes you a part of the problem. This must go at least double for security researchers.

One area where I feel as if there's some substantial overkill is in defining the scalability required of such a system. I think the authors bit off a larger than necessary chunk with an entire class B network. This may just reflect my lack of understanding of the problem and the way research is conducted in this field. It seems to me that working with much smaller chunks of IPs, in concert with researchers at other institutions would be the way to go. Using 5-10 physical machines to support VMs covering a /24 network seems more reasonable to me. Perhaps many of the same challenges remain.

I was fascinated by the role the gateway router plays. I would like to understand more about the way this system works. It seems to be a critical part of the puzzle, but its implementation seems to have been rather straightforward and no particular challenge to implement.

Containment while maintaining fidelity is, of course, a huge issue, and I think the approach
taken by these authors is quite reasonable—having the gateway reflect attacks/infection attempts back into the honey farm. This would be particularly reasonable given the large number of IPs available, something that wouldn't necessarily be true on a /24 network of IPs. Ahem!

As far as the implementation goes, I think I come away from my initial readings with a pretty good high-level understanding of the way things work in the prototype. The late binding idea is fascinating, allowing VMs to be instantiated on the fly, using flash cloning and delta virtualization, to stick around only as long as absolutely necessary. This is, of course, a requirement of the desired scale, and the efficiency of this process could be greatly reduced for a smaller scale effort. It was interesting to read the commentary on how “wasteful” honeypots are of computing resources, as they sit around waiting to be exploited. There's a lot of truth in this discussion, but isn't this the nature of honeypots?

I read with particular interest the section on limitations and challenges. The discussion on attracting attacks seemed to very much reflect the desire to scale up to such a high level. Smaller scale honey farms could make better use of transparent links to drive traffic to honeypots, as Michael and our group did quite successfully last fall. Michael apparently had equally good success with his high-interaction honeypot. There are some real problems with honeypot detection, some of which I heartily agree with. I think there are some additional measures that could be built into the honeypots to raise the bar on discovery, but this will always remain a problem, I think. The question in my mind is this: What percentage of the bad guys do we really need to fool, and how hard is that, really? My guess is that less than 5% of the working bad guys are sophisticated enough to really be a problem, and for these guys we’ll need to set up some number of non-virtualized systems. The problem will come in keeping the virtual farms good enough to fool most of the automated systems. Finally, the denial of service discussion is a good one. Again, I think this is a problem that potentially would affect the larger scale farms a lot more than it would smaller ones.