Dr. Curt Hastings
Remarks at November 9, 2018 Lexington Institute Federal IT Forum

Thank you. We've had some great talks that show us how cloud is not just transforming IT, but transforming everything about our online world. And I'm reminded that I have a file on my computer of projects not to do, because they are going to be AWS microservices in a few years. But I wanted to speak today about something that I haven't seen get much consideration. It is a problem that I see with the use of AI for cybersecurity.

The use of AI is superficially attractive because it addresses the main problem in defending our networks, which is cost imbalance, driven by better automation for the attackers. It would also address another major problem, which is that everything in our world is becoming an internet connected device.

Now maybe this bringing to mind the idea that AI will help the attacker more than the defender. And that's a valid concern. Recently a lot of work has gone into making AI robust to adversarial attacks, which is important in a domain like cyber. Some that come to mind are Bobby Filar's group at Endgame Systems, and the Windows Defender team. There is a synergy between ensemble methods and robustness, which is nice because Defender uses many learning subsystems, and it's being used in more and more places.

But I want to talk about a different problem. And to explain it, I need to review a little bit of the recent history of AI. Image recognition is everywhere in our online lives, but a lot of work from many very talented people was required before things got that way. About 10 years ago Fei Fei Li at Princeton created a collection of images called ImageNet, and people started running a recognition contest against it. About 4 years later Hinton's group at Toronto got a big performance jump in this contest with AlexNet, which opened a lot of eyes to the potential of deep learning architectures. All of this work has gotten the error rate to 1-3%, depending on how you frame the problem.

Why is this history important? AlexNet and its intellectual descendants solve one of the most worked on problems in AI. And the error rate has been stuck around 1%. It is 1% in the test environment, but it is probably worse in real life. That's a hard quantity to measure, but I don't think most of us would put the error rate on Google Image Search that low.

It turns out that is it very hard to achieve performance better than 1%, even in constrained environments. As things get messier and the goals to achieve become less well defined, the perceived performance of AI drops. At the risk of oversimplifying, I'll note that the AlphaGo frames the problem of playing Go in a way that makes the board look like an image, and that's also why I think we're farther from an AI solution to fake news than we would like.

So what does this all mean for our cybersecurity problems. Remember that the cyber problem is primarily one of scale. We have reverse engineers and hunt operators who are very good at what

they do, and they work very hard. But modern computers are extremely fast, and they're multiplying everywhere, so we have exponentially more data every year, but not exponentially more reverse engineers and hunt operators, even if we could afford to deploy them.

We might expect an AI to do a little better on the cyber problem than it does on the image problem, because the problem is more structured. If I could create an AI that flags all of the bad things on my network, and it has a real world performance of 98 to 99 percent accuracy, I can use it to halve the problem 6 times. That pushes the reckoning off a few years, but it doesn't erase it. My setup might be naive, but I hope this convinces you that we need to formulate the problem differently.

Fortunately, I don't think that the problem is intractable. I think a solution follows from two properties of computing. One is a property of the machines and one is of the humans.

First, the machines create a lot of data. And every performance advance in ML in some way relies on making use of more data. ImageNet was much bigger than the datasets that preceded it in the computer vision field. Modern computers are instrumented everywhere. Which brings me to the first thing that I hope you take from this talk. We should find ways to capture and use more of this data.

Second, humans are very good at a lot of tasks, but writing software isn't one of them. Many years ago the Department of Defense studied programmer productivity, and found that the average was about 24 lines of debugged code per day. You can do a lot worse than that, but empirically it's hard for a team to do better, and the productivity improvements that we see are mainly because we've moved to more expressive languages. That's a problem for the adversary too. There's no way that the amount of malware is as high as the estimates that we see. That's also true of the infrastructure and tactics and procedures that they use, since the modern way to define those things is also in software.

So if we can find a way to formulate the problem so we're only seriously considering things our adversaries create, and not things where they automated the creation, then the scale becomes manageable. To do that we need to be able to find similarities between the many variations of an underlying attack. It just so happens that in any consistent machine learning algorithm what we think of one example will depend only on what we know about the nearby examples. So this idea of similarity is probably on the right path, even if the paper that proved this property isn't widely known.

It turns out that all we need to do is catch enough of the bad things so that we have a few similar examples to anything that we might see anywhere on our network. How to do it is another talk, but I will tell you that similar programs will interact with the environment in similar ways, and the sensor data will be similar. And that there's enough data to see the similarity, as long as we make use of it.