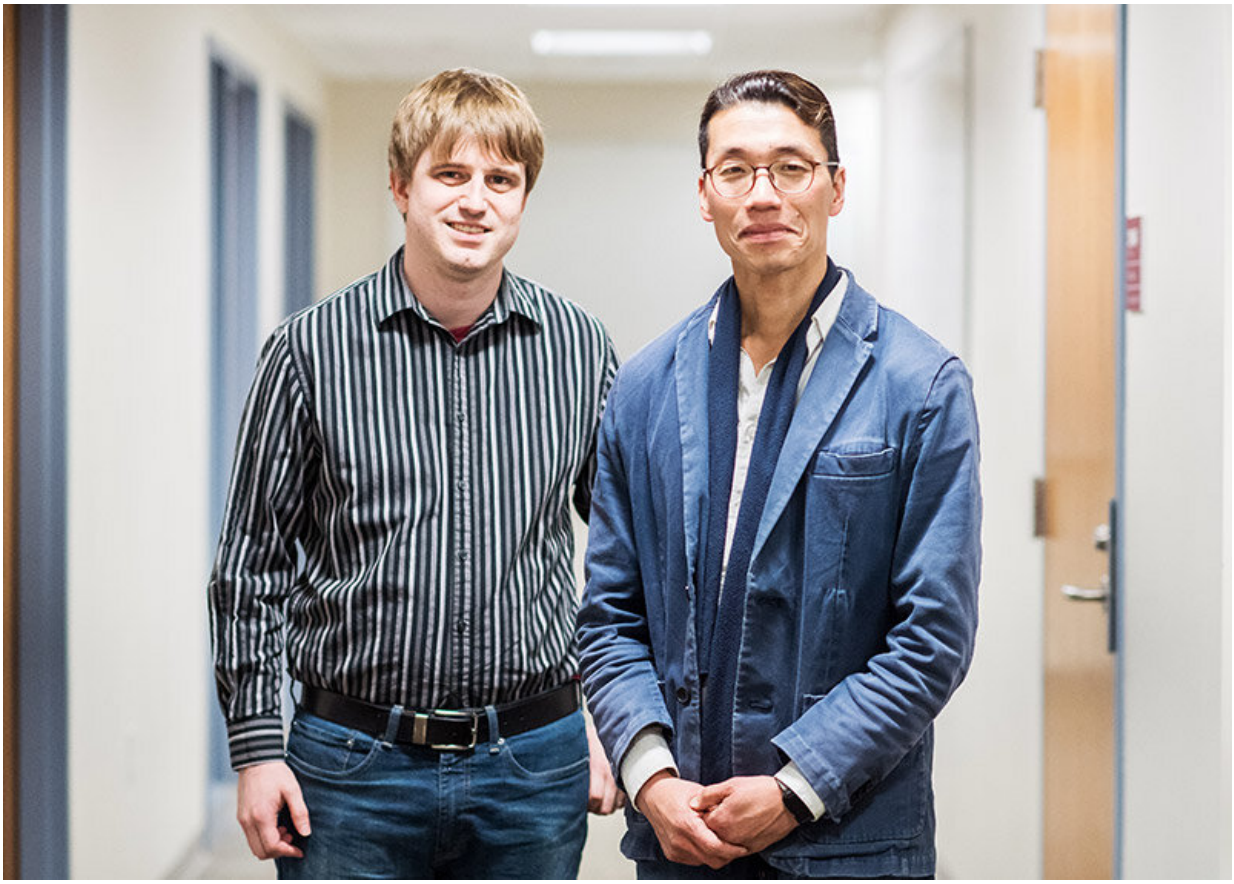
Could artificial intelligence make life harder for hackers?

February 1 2019, by Art Jahnke



PhD student Jacob Harer (left) and research professor Peter Chin worked with researchers from Draper to develop technology that could find the kind of software flaws that are often exploited by hackers. Credit: Jackie Ricciardi

As the volume of digital information in corporate networks continues to grow, so grows the number of cyberattacks, and their cost. One cybersecurity vendor, Juniper Networks, estimates that the cost of data breaches worldwide will reach $2.1 trillion in 2019, roughly four times the cost of breaches in 2015.

Now, two Boston University computer scientists, working with researchers at Draper, a not-for-profit engineering solutions company located in Cambridge, have developed a tool that could make it harder for hackers to find their way into networks where they don't belong.

Peter Chin, a research professor of computer science and an affiliate of the Rafik B. Hariri Institute for Computing and Computational Science & Engineering, and Jacob Harer, a fourth-year Ph.D. student in computer science, worked with Draper researchers to develop technology that can scan software systems for the kinds of vulnerabilities that are often used by cybercriminals to gain entry. The tool, which used deep learning to train neural networks to identify patterns that indicate software flaws, can scan millions of lines of code in seconds, and will someday have the ability to fix the coding errors that it spots.

Chin says the idea for the project, called DeepCode and funded by the DARPA (Defense Advanced Research Projects Agency) MUSE program and the Air Force Research Laboratory, came to him four years ago while he was delivering a lecture to his machine learning class (CS 542). Chin was describing the breakthrough achievement of scientists at Google and Stanford University, who used deep learning to teach a neural network to spot common patterns in millions of images and use the patterns to identify cats in YouTube videos. He wondered if a similar network could mine the big data of open-source programs and find patterns that indicate software vulnerabilities.

Chin knew that it was possible to represent a software program visually,

as a control flow graph. He also knew that there was a library of more than 10,000 common coding errors, called CWE (Common Weakness Enumerations), that had been put together by the National Institute of Standards and Technology (NIST). If those common coding errors in NIST's CWE could be presented as an image, he reasoned, a neural network could conceivably be trained on them to find common patterns of vulnerabilities, just as the Stanford neural network learned to identify common features of cats.

With that initial inspiration, Chin, who at the time was a chief scientist in decision systems at Draper as well as a professor at BU, helped secure funding for the project from DARPA. He, Harer (a Draper Fellow at BU), and colleagues at Draper began testing his assumptions on computer programs based on open-source C and C++ functions.

Since the start of the project in 2014, the researchers have realized that they needed more than just an image from the control flow graph to spot vulnerabilities. They have since improved their techniques, adding additional features, such as a parsed representation for code similar to that used by modern compilers, and they have adopted networks commonly used for natural language processing. Their research, which Chin says illustrates the promise of such university/industry partnerships, is now described in two papers, "Automated Vulnerability Detection in Source Code Using Deep Representation Learning," which has been accepted at IEEE ICMLA 2018, and "Learning to Repair Software Vulnerabilities with Generative Adversarial Networks," which was accepted at 2018 NIPS.

Chin says DeepCode's second function, fixing the coding errors, is still a working project. "It's very difficult," he says. "Correcting bad software is a lot like correcting bad grammar. Someone could say 'I went at the market' when they should have said 'I went to the market.' You train the network to identify the mistaken pattern and replace it with the proper

pattern. At least that's the basic idea."

Harer says one problem is that researchers don't know enough about how the machines recognize vulnerabilities. "These neural network models are very much black box models," he says. "They are trained on huge amounts of data and we sort of hope that they can figure out what's going on. This is a problem with deep learning in general."

Chin, Harer, and Draper researchers will continue to work on DeepCode, and plan to offer a version that can be deployed on a laptop and sent to corporations, most of whom are reluctant to share their code with outside parties, even for an examination that could save them tens of millions of dollars.

More information: Jacob Harer et al. Learning to Repair Software Vulnerabilities with Generative Adversarial Networks. arXiv:1805.07475 [cs.CL]. arxiv.org/abs/1805.07475

Rebecca L. Russell et al. Automated Vulnerability Detection in Source Code Using Deep Representation Learning. arXiv:1807.04320 [cs.LG]. arxiv.org/abs/1807.04320

Provided by Boston University