TEXT 2

# Researchers want to use hardware to fight computer viruses

## Date: November 7, 2016

Fighting computer viruses isn't just for software anymore. Binghamton University researchers will use a grant from the National Science Foundation to study how hardware can help protect computers too.

"The impact will potentially be felt in all computing domains, from mobile to clouds," said Dmitry Ponomarev, professor of computer science at Binghamton University, State University of New York. Ponomarev is the principal investigator of a project titled "Practical Hardware-Assisted Always-On Malware Detection."

More than 317 million pieces of new malware -- computer viruses, spyware, and other malicious programs -- were created in 2014 alone, according to work done by Internet security teams at Symantec and Verizon. Malware is growing in complexity, with crimes such as digital extortion (a hacker steals files or locks a computer and demands a ransom for decryption keys) becoming large avenues of cyber-attack.

"This project holds the promise of significantly impacting an area of critical national need to help secure systems against the expanding threats of malware," said Ponomarev. "[It is] a new approach to improve the effectiveness of malware detection and to allow systems to be protected continuously without requiring the large resource investment needed by software monitors."

Countering threats[[1]](#footnote-1) has traditionally been left solely to software programs, but Binghamton researchers want to modify a computer's central processing unit (CPU) chip -- essentially, the machine's brain -- by adding logic to check for anomalies while running a program like Microsoft Word. If an anomaly is spotted, the hardware will alert more robust software programs to check out the problem. The hardware won't be right about suspicious activity 100 percent of the time, but since the hardware is acting as a lookout at a post that has never been monitored before, it will improve the overall effectiveness and efficiency of malware detection.

"The modified microprocessor will have the ability to detect malware as programs execute by analyzing the execution statistics over a window of execution," said Ponomarev. "Since the hardware detector is not 100-percent accurate, the alarm will trigger the execution of a heavy-weight software detector to carefully inspect suspicious programs. The software detector will make the final decision. The hardware guides the operation of the software; without the hardware the software will be too slow to work on all programs all the time."

The modified CPU will use low complexity machine learning -- the ability to learn without being explicitly programmed -- to classify malware from normal programs, which is Yu's primary area of expertise.

"The detector is, essentially, like a canary in a coal mine to warn software programs when there is a problem," said Ponomarev. "The hardware detector is fast, but is less flexible and comprehensive. The hardware detector’s role is to find suspicious behavior and better direct the efforts of the software."

Much of the work -- including exploration of the trade-offs of design complexity, detection accuracy, performance and power consumption -- will be done in collaboration with former Binghamton professor Nael Abu-Ghazaleh, who moved on to the University of California-Riverside in 2014.

Lei Yu, associate professor of computer science at Binghamton University, is a co-principal investigator of the grant.

Grant funding will support graduate students that will work on the project both in Binghamton and California, conference travel and the investigation itself. The three-year grant is for $275,000.

## Reading Comprehension Tasks

**1) What is Ponomarev's project called?**

Practical Hardware-Assisted Always-On Malware Detection.

**2) Why might his project have national impact?**

Rather than national, the impact would be worldwide since it will potentially be felt in all computing domains, from mobile to clouds.

**3) Explain in what way malware is increasing and also its consequences.**

More than 317 million pieces of new malware -- computer viruses, spyware, and other malicious programs -- were created in 2014 alone, according to work done by Internet security teams at Symantec and Verizon. Malware is growing in complexity, with crimes such as digital extortion (a hacker steals files or locks a computer and demands a ransom for decryption keys) becoming large avenues of cyber-attack.

**4) According to the researchers, what happens once an anomaly has been detected?**

"The detector is, essentially, like a canary in a coal mine to warn software programs when there is a problem," i.e., the alarm will trigger the execution of a heavy-weight software detector to carefully inspect suspicious programs. The software detector will make the final decision.

**5) According to the project, the hardware detector is not 100% accurate, so how is its effectiveness explained? Explain the procedure.**

The hardware detector’s role is to find suspicious behavior and better direct the efforts of the software.

**6) Explain the phrase "acting as a lookout"**

A lookout is a high place where a person can look at what is happening in the area around them, especially in order to watch for any danger. This project is not intended to have a direct struggle against malware but better warning the pertinent software to act accordingly.

**7) Explain the comparison "like a canary in a coal mine". Find information in the Internet and connect it with the text.**

This is connected with the previous answer. It is well known that it was a custom in mining to carry a canary where humans had to work. Any canary is very susceptible to gas emanation, leak that represents an enormous danger to humans since gas naturally has no odor, i.e., humans can’t smell a gas leak. The moment the canary was falling dead was the moment to rush to the surface.

TERMINATOR THEORY

I chose this article because it refers to a story that was written thirty years ago. A situation that is related to technologies that do not yet exist and that, at first, fascinate us, and then we assume we will not get along with.

But, let us start at the beginning.

This article is some news about a project that is currently being developed with the collaboration of two teams of researchers, one from the University of Binghamton and the other from the University of California-Riverside.

This project consists of modifying a microprocessor that, with artificial intelligence hardware code, can make an early detection of malicious software. Without a doubt, of all the solutions proposed, this is the best one. The response time of a set of instructions governed by hardware is exponentially higher than any software we could ever achieve.

The article provides some more details that are essentially trivial. What has been said so far is a good summary. We do not know if they have made any concrete development yet (the article is 5 years old), and if they have, at what stage they are. There are no articles on the internet that expand the information, much less do we have news that the main industrial developers of microprocessors have been involved in any way.

So therefore, the project itself has probably been frozen without significant progress. But the idea has already been put on the table, and although it is an innovative and critical idea, we may not be at a sufficiently advanced technological stage to implement it. And the latter, more than an assumption, is a deduction with enough evidence.

Let me explain myself. Here at the university, we study (as part of the curriculum) the so-called electronic gates governed by a binary logic from which microprocessors are molded. We have even built, theoretically, a 4-bit microprocessor capable of running very simple programs. These programs, completely printed on logic gates, perform operations that are linear with no heuristic capabilities.

Knowing the state of the science behind this, the proposal of an artificial intelligence program embodied in hardware is not that it is impossible, but it is flatly unlikely.

However, what is striking is that the idea is not new. It was raised 30 years ago. And I want to draw the attention to the reaction of humanity the first time that listened to the idea. And that is the comment I want to make in relation to this text.

Before, let's go back 4 decades, to the 80's. In 1984 the movie Terminator was released. This movie, was it a science fiction or a horror movie? People who have debated this subject have concluded that it was a mixture of both. The science fiction side does not concern us right now to what we are now dealing with. The interesting side is that the film was considered a horror movie. Terror of what? The plot of the film was close to subject that, without being bloody, produced terror in those years. The cold war, for example. But, at a level closer to the relationship that humanity was beginning to have with machines, in this relationship, were we outmatched. And eventually, controlled?

A micro argument within the script is enough for this. Let's see:

In one of the first scenes, you can hear the answering message from Sarah's (the protagonist) answering machine: "Hello. I fooled on you. You're talking to a machine. But don't be shy. Machines need love too, so talk to it."

The dark reverse is given in one of the last scenes. When Sarah calls her mother, she ends up talking to the Terminator, who mimics her mother's voice. She is talking to a machine, but now she is the one being fooled believing that she is talking to a person.

That background of the movie that takes us to its sequel in the 90's, 1991 to be exact, now 30 years ago. In that sequel, the terrifying element of the first one is captured in a more patent way by the hand of one of its protagonists, Dr. Miles Bennett Dyson. He is the original inventor of the microprocessor that would lead to the development of Skynet, an intelligent computer system destined to be controlled by the United States military, but which would later launch a global war of extermination against humanity.

In one part of the movie, Dyson claims that the microprocessor they were studying (which was recovered from the Terminator from the first movie) "It was a scary thing, radically advanced. It was smashed ... it didn't work. But it gave us ideas. It took us on new directions ... things we never would have thought of. All of our work is based on that. "

He was referring to a microprocessor that had innate artificial intelligence capabilities…

I have always been amazed by the insight of the writers of the original script of the film, because it is possible to find details of that "terrifying technology" that comes to the case of what we are speaking today: the characteristics. And this science fiction movie can give us an idea of ​​the state of the art in front of the news we are now considering.

What are these characteristics, those of a microprocessor with innate artificial intelligence capabilities?

According to the writers, after many years of work, Dyson finally came up with a prototype model that not only worked and was bug-free, but was superconducting at room temperature, making it extremely powerful.

The CPU was primarily modeled and designed on computers in advanced three-dimensional programming packages, where simulated tests could be performed in real time.

The lattice of cubes in the construction of the microprocessor prototype suggests a "hypercube" (a cube of more than three dimensions). In microprocessor design, hypercubes are used as a physical connection scheme that minimizes the effective communication distance, and therefore, time delay between sub processors when the logical connection scheme needed by the software to be run on those processors cannot be known in advance. This then supports the neural network's ability to learn, adapt, and build new logical connection schemes.

But the most important part is that the neural network is innovative and uses the latest quantum (ternary) effect chips. Until Dyson designed the microprocessor, computers could run on chips made up of millions of transistors. Traditional computing was done in the current (binary) system: ones and zeros, on and off.

That is, this microprocessor made it possible to do many more calculations every second, with possible trillions of switching positions, many of them simultaneous at each quantum level.

And so far, those who knows the subject, knows that this, in our current stage of technology, exists, but not in a concrete way. We don't even have programming languages ​​capable of taking advantage of our current multi-core microprocessors (we still use a single core). And if this already would have a solution, we still need one more: a programming language with ternary capacity, a programming language that currently simply does not exist yet.

It is all this that explains the radio silence of this for five years now, and the absence of news about this project, which simply does not have a current technological advance that can make real the proposal.

Later will come the debate of what to really expect from this advance. And even more critical: if this does not would end up being the beginning of a more feared Monster of Frankenstein.

intel threat detection technology encounter

STICKY NOTES:

* 2006 – article about the investigation of Ponomarev about artificial intelligence embedded into microprocessors
  + At a first moment, there were 5 years of radio silence
  + Was a stolen idea from terminator the movie?
  + **GO TO 2021**
* 2018-2019 - found new paper by Ponomarev about side-channel attack
  + It was other side of the same coin? Seemed similar, but it was not exactly the same
  + What’s going on? He dropped out the former project?
  + **GO TO «PUTTING ALL TOGETHER»**
* 2021 – Intel (integrated electronics) releases threat detection technology
  + I’ve found news about new release from intel. Reading about it, I was shocked: was the same idea of Ponomarev. What’s going on?
  + Searching again for information about Ponomarev, original proposal felt into oblivion
  + **GO TO 2018-2019**

PUTTING ALL TOGETHER

1. Contrarrestar las amenazas. [↑](#footnote-ref-1)