Project 5: "Cyber Attack and Defense"

Area Coordinator:
Dr. John Franco
Geier Professor, Computer Science
Department of Electrical Engineering and Computing Systems (EECS)
College of Engineering and Applied Science
PO Box 210030
University of Cincinnati
Cincinnati, OH 45221-0030
Office: 831 Rhodes Hall
E-Mail: franco@gauss.ececs.uc.edu
Phone: 513-556-1817

Sub-Area Coordinator:
Dr. Paul Talaga
Assistant Professor – Educator, Computer Science
Department of Electrical Engineering and Computing Systems (EECS)
College of Engineering and Applied Science
PO Box 210030
University of Cincinnati
Cincinnati, OH 45221-0030
Office: 806B Rhodes Hall
E-Mail: talagapl@ucmail.uc.edu
Phone: 513-556-4744

Graduate Research Assistant:
Mr. Carlo Perottino
Graduate Student in Computer Science and Engineering (CSE)
Office: 831 Rhodes Hall
E-Mail: perottca@mail.uc.edu
Phone: 937-748-9696

Project Summary

It is well-known that attacks on industrial, government, and academic digital installations via the Internet threaten the security and economy of the United States. While most attacks are amateurish and can be detected fairly easily, the more serious attackers are continually changing their tactics to be harder to defend against. They may be trying to steal Intellectual Property, or trying to damage an organization's infrastructure and/or data, or they may just be interested in stealing currency. These attackers are the dangerous ones.

A number of technological devices have been developed over the years to ensure confidentiality, data integrity, and authenticated communications. Notable examples include various public key cryptosystems such as RSA, Diffie-Hellman key exchange and elliptic curve versions of these, secure symmetric key cryptosystems such as AES-256, and hash-based systems for checking the integrity of data. Unfortunately, attackers have developed ways to side-step the security of these cryptosystems with side-channel attacks using timing or differential power analysis, or social engineering, or weaknesses in Operating System design and implementation, or by exploiting vulnerabilities that inadvertently exist in code due to bugs introduced by coders or incomplete testing against requirements. Attacks have become so successful that many people wonder whether it is possible to secure the Internet at all.

This project aims to shed some light on why it seems so difficult to protect the Internet. Using tools for attack, participants will see how an attack is planned and launched. Participants will see that effective attacks do not happen instantaneously but occur over a long period of time in phases that are characterized as a cyber kill chain. In the first, reconnaissance phase, the attacker gets some idea of the victim's network topology and potential vulnerabilities. In the second, weaponization phase the attacker has crafted a tool for attack based on the information gathered in the first phase. Later stages involve delivery of the attack to the victim, resulting in the installation of the attacker's software, and finally execution of the attack. What has people really worried is

that in some cases the execution is delayed indefinitely, possibly until the moment that a massively destructive cyber event is initiated, possibly by a foreign government. Unfortunately, it is often very difficult to detect a dormant attack payload as it generally does not give any indications of being active. What can be done, if anything?

For one thing, breaking the chain before execution will prevent or at least mitigate damage. There are numerous tools for doing this and project participants will experiment with some of them. But, detection and prevention tools cannot be static because sophisticated attackers will modify how they do reconnaissance so they look like benign traffic, for example. So, existing defensive tools eventually have to be modified or replaced. How is this done?

In this project the teachers will learn research tools which include offensive tools and defensive tools - understand what these accomplish and how they do it run both kinds of tools to develop an appreciation of the power and limitations of each

Possible Ideas for Classroom Implementation

We can consider the big idea that electronic computing and communication pose some of the most complex challenges engineering has ever faced. These challenges range from protecting the confidentiality and integrity of transmitted information and deterring identity theft to preventing the scenario dramatized in the Bruce Willis movie *"Live Free or Die Hard,"* in which hackers take down the transportation system, then communications, and finally the power grid. An essential question that can be considered related to this big idea for an attack that appears to be impervious to existing tools, can a new tool be designed to break the kill chain? The challenge to be pursued is development of this tool or app. One can consider a wide array of guiding questions that can aid in creating the solution of this challenge, which can be grouped in four broad categories: how attacks are executed; how known attacks are prevented; how unknown attacks are detected; how damage is mitigated if an attack succeeds.

Possible Activities that can be undertaken as a response to the guiding questions:

1. Understand how attacks are executed and detected:
   a. Install VirtualBox on a Windows or Mac computer
      Download kali linux and security onion OS install iso
      Install kali and security onion as virtual machines in VirtualBox
   b. Learn how to use metasploit, armitage, nmap, wireshark and other attack tools
      Learn how to use squil, squert, elsa, snorby, (and wireshark) and other defensive tools
2. Attack and defend:
   a. Perform attacks on security onion from kali
      Use a variety of attacks, especially on applications like ftp, web servers, adobe acroread, etc.
      Watch the attacks unfold using the defensive tools, logs, etc.
   b. Prevent, mitigate, or recover from damage due to the attack using existing tools
      Define some quality and deployment of services to protect
      Harden the OS against attack – proper configuration of the services
      Reconfigure net topology after attack detection – block IP addresses
      Prevent or mitigate denial of service (DoS) attacks
3. Make detection harder and try again:
   a. Inject a lot of benign traffic into the “network” while attacking
      How difficult is it now to detect and mitigate?
   b. Find or develop an attack that seems to be undetectable using current tools with much benign traffic
4. Create a strategy for detecting the attack in 3b and killing the attack before it is executed.

Note: By using virtual machines, damage to the host is prevented.