Global Information Assurance Certification Paper

A Brief Overview of Software Agent Applications and Risks
Stephen V. Pellissier
November 10, 2000

Introduction

This paper gives a brief description of software agents, then describes a sampling of evolving agent applications that apply to information security and points out some of the risks incurred by employing agents, along with possible mitigation strategies.

In the evolution of software development methodologies, starting with machine and assembly language and progressing through functional and object-oriented approaches, agents are one of the latest developments [9]. A software agent is a tool that inhabits a computer or network assisting applications and users accomplish tasks. Because agents may be required to execute on heterogeneous computer systems, they are often written in an interpretive script/language, such as Java [3]. The idea of software agents is not new, but their widespread application was limited until the explosive growth of the World Wide Web with Java enabled web browsers seemed to excite interest [4].

Agents differ from other software in a number of ways. Some of their more distinguishing characteristics include that they can be: *autonomous*, meaning they can act independently, without human intervention; *proactive*, meaning they are goal-oriented, capable of taking the initiative, rather than just reacting to the environment; *responsive*, meaning they can anticipate the user's needs in a changing environment; and *adaptive*, meaning they can learn their user's preferences and adjust accordingly [3]. Agents can also be customized to serve in a specific role or accomplish a specific mission. Static agents reside on one platform, while mobile agents can move throughout a network, executing on one platform, then halting execution and moving to another, then re-starting execution.

Applications

Agents have many uses in the real world. Jennings and Wooldridge identify a number of applications for using agents ranging from process control to air traffic control to information gathering/filtering to patient monitoring and computer games [5]. This paper will, however, focus on two developing agent applications pertaining to information security.

Vulnerability Scanning

Most vulnerability scanning tools currently available (e.g., ISS, Nessus, Nmap, etc.) are standalone packages that search for known vulnerabilities. Humphries et al. describe a mobile agent design, whereby various agents move through the network looking for specific vulnerabilities on each host [2]. These mobile agents consist of three parts: an itinerary section, which outlines the agent's path through the network; a code section, which the agent executes upon arrival at a host; and a payload section, which contains a host by host diary of the results (i.e., identified vulnerabilities) of the agent's execution. Humphries asserts that the advantages of using mobile agents over more traditional tools are first that the agents can be quickly updated when new vulnerabilities are discovered, second that network growth can be easily handled by simply adding more agents.

Intrusion Detection and Intrusion Response

Jansen et al. identify a number of functional and performance requirements for an intrusion detection system, but in general, intrusion detection system should continuously monitor network behavior, be fault tolerant (since it too may be the target of attack), be capable improving its detection capability by adapting to changes and receiving updated attack signatures, and accurately (i.e., with low false alarm rate) report anomalies or possible intrusions, supplying adequate information to deal with an intrusion [4].

Ragsdale et al. proposed a system of agents for use in intrusion detection. The Adaptive Agent-based Intrusion Response System (AAIRS) provides response adaptation by giving a relative weighting to different response techniques [6]. Responses that have previously been successful receive greater weight and are therefore used more often than less successful techniques. At the center of this system is a master analysis agent, whose purpose is to determine whether an attack is new or a continuation of a

previous incident [1, 6]. The master analysis agent obtains attack information via an interface agent, which deals directly with the intrusion detection system(s). If an attack is a continuation of a previous incident, the master analysis agent passes the attack information to an already existing analysis agent, who handles the incident until it is resolved. If it is a new attack, the master analysis agent creates a new analysis agent. The analysis agents rely on input from other agents to develop a course of action that is based upon policy, which can then be decomposed into the specific actions that must be taken in response to the attack.

Risks

While agents can serve in a variety of information security support roles, security risks associated with the agents themselves may impede their acceptance and widespread application. If, for instance, a trusted agent is compromised the entire network it was designed to protect could be at risk. Someone attacking the network could gain valuable vulnerability information by observing which agents are traversing the network and what data they have collected. The agents' code and/or data could also be changed. As a result, the agents could malfunction, report incorrect data, or even damage the systems they were designed to protect.

Jansen identifies three main threat categories [3]:

- Agents Attack Platforms. Agents could attack the agent platform (i.e., the agent's computational environment, such as a trusted host computer) by either gaining unauthorized access to information stored on the platform or executing in a disruptive manner, such as exhausting computational resources or even shutting the system down.
- Platforms Attack Agents. Alternatively, the agent platform could attack the agent. In this case the platform could take actions ranging from denying the agent access to required services, to copying, modifying, and/or corrupting an agent's code and/or data, to terminating the agent. Not only could malicious modification of an agent's code damage an agent, but it could also serve as an avenue for attacking other platforms.
- Agents Attack Agents. Agents could monitor other agents' actions and launch attacks to interfere with an agent's mission, intercept its service calls and reporting mechanisms, steal or modify its data and/or code, and terminate it by causing unhandled exceptions.

Risk Mitigation

Encryption schemes have been recommended to protect the confidentiality of the agents' code and data while the agent is in transit [2, 3]. Likewise, digital signatures can protect the integrity of the agent's code, data, and itinerary or path. During execution of the code, however, encryption cannot protect the agent since its code must be in clear text to execute [2].

Jansen describes 13 countermeasures, seven of which deal with detection mechanisms (three applicable to the agent platform and four applicable to the agent), while the remaining six deal prevention mechanisms (three for the platform, three for the agent) [3]. Below is a description of two such countermeasures:

- Path History deals with detection at the agent platform. This countermeasure involves each agent platform adding a non-reputable entry in the agent's path history. Each newly visited platform can then authenticate the record of prior platforms visited and determine whether to trust the agent.
- Partial Result Encapsulation deals with detection for agents. Using a public key to encrypt the data at each agent platform, encapsulated data can be incrementally accumulated until the agent returns to its reporting location, where the private key is used to peel away the layers of data. By employing encryption and digital signatures, encapsulation of the results of an agent's visit to each host can, respectively, provide confidentiality and integrity.

Summary

The use of agents in the information security arena is growing. This paper briefly described the development of using agents for vulnerability scanning, intrusion detection, and intrusion response. Agents can offer system administrators another tool for managing and protecting networks as they

become larger, more distributed, and more diversified. Schwartau describes protection of information assets in terms of detection and response time [7]: The faster a system can detect and respond to an intrusion the more "protected" it is. Response time could also improve by using agents to assist in the administrator's decision whether or how to respond to an attack. Along those lines, Surdu proposed the use of agents to monitor military operations, comparing the real operations to a simulation, and providing advise the commander [8].

By helping the administrator identify vulnerabilities, detect and respond to attacks, and assist in determining a course of action, the use of agents can be a valuable tool to enhance network security. At the same time, system administrators must be aware that agents do not come without risk. The fact that they can be captured, then read and/or tampered with should be considered before taking drastic action in response to the data they report.

References

1. **Carver, Curtis A., et al. "A Methodology for Using Intelligent Agents to Provide Automated Intrusion Response." IEEE Systems, Man, and Cybernetics Information Assurance Workshop, West Point, NY, 6-7 June 2000. URL: http://www.itoc.usma.edu/surdu/ (10 November 2000).**
2. **Humphries, Jeffrey W., et al. "Secure Mobile Agents for Network Vulnerability Scanning." IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop, West Point, NY, 6-7 June 2000. URL: http://206.96.207.5/curt/ (10 November 2000).**
3. **Jansen, Wayne A. "Countermeasures for Mobile Agent Security." National Institute of Standards and Technology. URL: http://csrc.nist.gov/mobileagents (10 November 2000).**
4. **Jansen, Wayne, et al. "Applying Mobile Agents to Intrusion Detection and Intrusion Response." National Institute of Standards and Technology. NIST Interim Report – 6416. October 1999. URL: http://csrc.nist.gov/mobileagents (10 November 2000).**
5. **Jennings, N. R. and M. Wooldridge. "Applications of Intelligent Agents." in Agent Technology Foundations, Applications, and Markets. Springer-Verlag, 1998. URL: http://agents.umbc.edu/introduction (10 November 2000).**
6. **Ragsdale, Daniel J., et al. "Adaptation Techniques for Intrusion Detection and Intrusion Response Systems." IEEE International Conference on Systems, Man, and Cybernetics, Nashville, TN, 2344-2349. 8-11 October 2000. URL: http://206.96.207.5/curt (10 November 2000).**
7. **Schwartau, Winn. *Time Based Security*. Interpact Press, Seminole, FL. 1999.**
8. **Surdu, John R. and U. W. Pooch. "Rational Agents, Simulation and Military Operations." Proceedings of Artificial Intelligence, Simulation, and Planning in High Autonomy Systems, AIS2000, Tucson, AZ. 6-8 March 2000. URL: http://www.itoc.usma.edu/surdu (10 November 2000).**
9. **The Open Agent Architecture. URL: http://www.ai.sri.com/~oaa/ (10 November 2000).**

Last Updated: July 23rd, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor