MANAGEABLE NETWORK SECURITY SYSTEM DETECTION USING MOBILE AGENT

Wawan Indarto
Cisco Networking Academy Program
Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia
Kampus Terpadu UII, Jl Kaliurang Km 14.5 Yogyakarta
E-mail: wawan@fti.uii.ac.id

Abstract

Protection of agent data state and partial results in mobile agent systems continues to draw research interest. Today hackers disguise their attacks by launching them form a set of compromised hosts distributed across the Internet. It is very difcult to defend against these attacks or to track down their origin. Commercially available intrusion detection systems can signal the occurrence of limited known types of attacks. New types of attacks are launched regularly but these tools are not effective in detecting them. Human experts are still the key tool for identifying, tracking, and disabling new attacks. Often this involves experts from many organizations working together to share their observations, hypothesis, and attack signatures.

.

Keyword*: agent, intrusion detection, network*

Over the years computer systems have evolved from centralized monolithic computing devices supporting static applications, into client-server environments that allow complex forms of distributed computing. Throughout this evolution limited forms of code mobility have existed: the earliest being remote job entry terminals used to submit programs to a central computer and the latest being Java applets downloaded from web servers into web browsers. A new phase of evolution is now under way that goes one step further, allowing complete mobility of cooperating applications among supporting platforms to form a large-scale, loosely-coupled distributed system. The catalysts for this evolutionary path are mobile software agents – programs that are goal-directed and capable of suspending their execution on one platform and moving to another platform where they resume execution.

More precisely, a software agent is a program that can exercise an individual's or organization's authority, work autonomously toward a goal, and meet and interact with other agents. Possible interactions among agents include contract and service negotiation, auctioning, and bartering. Agents may be either stationary, always resident at a single platform; or mobile, capable of moving among different platforms at different times. The reader is referred to for a more extensive introduction to the subject. For example, a simple agent written as a Java applet has mobility of code through the movement of class files from a web server to a web browser. However, no associated state information is conveyed. In contrast, Aglets, developed at IBM Japan, builds upon Java to allow the values of instance variables, but not the program counter or execution stack, to be conveyed along with the code as the agent relocates. For a stronger form of mobility, Sumatra, developed at the University of Maryland, allows Java threads to be conveyed along with the agent's code during relocation. The mobile agent systems under discussion in this paper involve the relocation of both code and state information. Mobile agents are a method for implementing distributed systems and an emerging abstraction for architectural program design. Unresolved security issues continue to be cited among other reasons why mobile agents have not achieved widespread implementation .Research continues to formulate mechanisms that secure the computational state of mobile agents in the presence of malicious hosts.

Protecting an agent consists of two primary features: protecting the static executable code of an agent from disclosure or alteration and protecting the dynamic state of the agent as it incrementally changes during execution. In this project we recognize that human experts will remain a critical part in the process of identifying, tracking and disabling computer attacks. We also recognize that an important part of the discovery, analysis, and defense against new distributed attacks is the cooperation that occurs between experts across different organizations. Many installations do not have the expertise necessary to develop full attack analyses. Our goal is to build automated tools for computer experts and system administrators to:

1. Identify the characteristics of an attack given data from network sensors
2. Develop a hypothesis about the nature and origin of the attack
3. Share that hypothesis with security managers from other sites
4. Test that hypothesis at those other sites and coordinate the results of testing
5. Archive the data necessary for use as evidence in later law-enforcement actions

The main diifficulties in catching the bad hackers arise because[2]

1. Existing system logs are too large;
2. Existing system logs do not contain all the information that could be useful in tracking down anintruder
3. Attacks often come from multiple sources and spawn processes at multiple destinations but there is no way of coordinating logs across administrative domains.

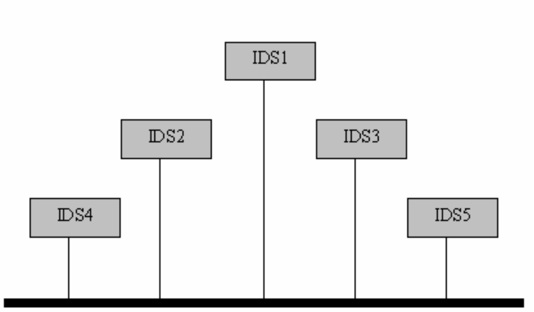

Figure 1. Uncoordinating IDS

We believe that mobile agent systems can play a key role in addressing these diffculties. Because formulating a correct hypothesis requires coordinating and correlating system logs from multiple locations, mobile agents are a well-suited processing paradigm. A mobile agent can travel to the location of the system logs, filterlter out the time-relevant or location-relevant information from these logs, and correlate all this information without necessitating costly transfers across the net. We envision a new generation of security systems that work across administrative domains and use mobile agent technology to share and correlate system logs. We believe it is feasible to organize groups of computers that have different administrative domains as logical entities and use a mobile agent system to access, share, and correlate its system logs. Such alliances are already becoming a reality. For example, the Information Technology *Information Sharing and Analysis Center* (IT-ISAC) is a consortium of 19 technology giants including *Microsoft* and *Oracle* who oined forces recently to share security data, to protect hemselves from hackers by sharing reports of electronic threats, incidents, solutions and countermeasures.

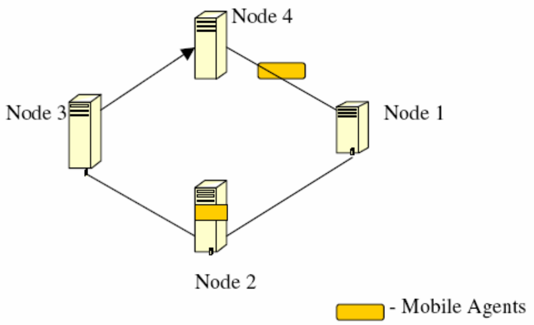

Figure 2. Using mobile agent to coordinate IDS

We believe that mobile agent systems can play a key role in addressing these diffculties. Because formulating a correct hypothesis requires coordinating and correlating system logs from multiple locations, mobile agents are a well-suited processing paradigm. A mobile agent can travel to the location of the system logs, filterlter out the time-relevant or location-relevant information from these logs, and correlate all this information without necessitating costly transfers across the net. We envision a new generation of security systems that work across administrative domains and use mobile agent technology to share and correlate system logs. We believe it is feasible to organize groups of computers that have different administrative domains as logical entities and use a mobile agent system to access, share, and correlate its system logs. Such alliances are already becoming a reality. For example, the Information Technology *Information Sharing and Analysis Center* (IT-ISAC) is a consortium of 19 technology giants including *Microsoft* and *Oracle* who oined forces recently to share security data, to protect hemselves from hackers by sharing reports of electronic threats, incidents, solutions and countermeasures.

The mobile agent security system can be built on op of any mobile agent system, such as our own DAgents system. The main component of such a system would be[1]

1. The basic agent infrastructure
2. Module for the distributed capturing and accessing of security logs
3. A module for correlating data from the security logs
4. A module for formulating at tack hypotheses based on the output of the correlation module.

Agents wishing to search the collection may jump to any one of the hosts, connect to the document index, and query the index for relevant documents. Then, they can choose to read the documents through an internal Network File System or by jumping to each host to examine the documents locally. The goal is to examine issues in constructing scalable mobile-agent services, balancing load among the hosts, and allowing the agents autonomy in deciding how best to service their query. We plan to use Serval for intrusion detection by us ing the same infrastructure to store network and host log data. The database can index the logs for quick searching. Network administration tools can use mobile agents to examine the data collection efficiently and exibly, because complex analyses can be encoded as mobile agents, sent to the Serval cluster, and examine the data with high-speed local access to the data.

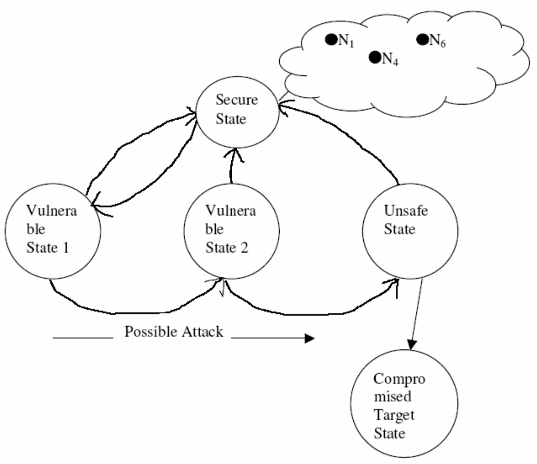

Figure 3. Threat in network

We plan to further extend the Serval infrastructure beyond its specialized cluster. In a local-area network, intrusion-detection data is inherently collected at distributed sites, at the hosts or network segments where intrusions may occur. The log data is stored in lo cal hosts (possibly specialized secure logging hosts), avoiding the need to transmit it to distant parts of the network. By installing the Serval software on these log hosts, we create a virtual Serval cluster" as a distributed network service. Analysis tools can launch a mobile agent into any one of these log hosts, and the mobile agent can search the distributed index and jump to other log hosts as needed to examine the log data. Consider a typical network intrusion. After a reconnaissance phase (i.e., the hacker scans a certain network to and hosts that exhibit a given vulnerable service), a vulnerability is exploited, for example executing a buffer overow attack against a machine with ftp enabled. This allows the hacker to overcome the standard login process and gain root privileges on the compromised machine. The hacker, then, performs operations on the compromised machine (read/modify data,install backdoors) and tries to compromise further internal machines of the same organization. These further attacks are likely to be easier than the first one,since internal machines are often less protected than the network's perimeter. Trust relationships, shared file systems, weak passwords are examples of common internal weaknesses that an hacker could easily exploit.

Given this typical scenario, the Serval mobile agent infrastructure is useful for the following goals. In the incident handling phase, it can help identify all the operations performed by the attacker and their time sequence. Serval is necessary to completely restore the compromised systems and avoid further attacks. In aforensic analysis, it supports the correlation among logs from different machines and the retrieval of evidence of the attack distributed among diferent machines of the network, which is useful for proving the damages caused by the attack. If a response strategy is considered, the Serval infrastructure can be used to stop the intrusion at its early stages, by isolating compromised machines. Such a response is necessary since today it is unfeasible to assume that all attacks could be blocked at the network perimeter.

References

[1] J. Aslam, K. Pelekhov, and D. Rus, *Apractical clustering algorithm for static and dynamic information organization*, Proc. of the 1999 Symposium on Discrete Algorithms.

[2] R. Gray, D. Kotz, S. Nog, D. Rus, S.Chawla, and G. Cybenko, *Agent Tcl: Targeting the needs of mobile computers*, IEEE Internet Computing, 4(1) 58-68, July-August 1997.

[3] D. Rus, R. Gray, and D. Kotz, *Transportable Information Agents*, Intelligent Information Systems, vol 9. pp 215-238, 1997.

[4] S. Northcutt, and J. Novak, *Network Intrusion Detection: An Analyst's Handbook*, 2nd Edition, New Riders Publishing. 2000.

[5] S. Northcutt, M. Cooper, M. Fearnow, and K. Frederick, *Intrusion Signatures and Analysis,*New Riders Publishing, February 2001.

[6] J. Scambray, S. McClure, and G. Kurtz ,*Hacking Exposed*, 2nd Edition, McGraw Hill, October 2000.

[7] Indarto, Wawan, *Lecture Handout : Computer Security System*, Informatic Department, Islamic University of Indonesia, 2003.

[8] Cisco Inc, *Bridge Course Material CCNA V3.1*, Juni 2005.