Achieving Near Real-time Security Using an Intelligent Agent "Snapshot" Approach

Dr. Bruce C. Gabrielson, NCE
gabrielson_bruce@bah.com
Booz Allen & Hamilton
Linthicum, MD

Mr. Leonid Kunin
Leonid.Kunin@saic.com
Center For Information Security Technology
Science Applications International Corporation
Columbia, MD

Abstract

This paper describes an approach using intelligent agents to enhance security in large networks. In particular, the security of large networks is difficult to manage in the face of changing technologies, limited resources, and ever increasing threats. Real world realities now dictate quick efficient solutions just to keep the network functioning. The current emphasis on reactive security is being replaced with an urgent need for immediate detection and automated corrective action. Using an agency (intelligent agents) to address near real-time "snapshots" of configuration changes, remote network testing, and remote automated configuration tools, a full near teal-time detection and reaction capability is possible. The combination of Government and SAIC-developed Trover, NTSecWiz, PConfig, and Situation Response Agent (SRA), along with other currently available and emerging tools, represents an integrated higher level agency that may achieve this capability.

Introduction

Science Applications International Corporation (SAIC) is a diversified high-technology research and engineering company based in San Diego, California. SAIC offers a broad range of expertise in technology development and analysis, computer system development and integration, technical support services, and computer hardware and software products.

Significant research efforts have been put forth by SAIC's Center for Information Security Technology (CIST) and others to identify viable methods for efficient and efficacious security related to information systems. New and innovative measures are required to enable enclave or virtual clients to use the Internet and still protect critical data, regulate dissemination, guard against malicious behavior, and ensure the integrity of the system without impeding normal activity.

This paper addresses a near real-time security improvement approach related to enterprise-wide computing using intelligent agents. Specifically, the paper will address:

- Significant reductions in human intervention using an integrated agent-based design approach.

- Agency applications that address defense-in-depth.

- A snapshot approach to produce meaningful near-real time configuration management conditions in large network applications.

- Local processing of large data sources with agent indicator communications between sources and receptors.

- Interactive data mining to allow authorized users to identify and access related information for in-depth analysis across heterogeneous data types.

Information Assurance Problems

In the real world there are practical problems with information assurance. Technology advances result in equipment and configuration changes that are increasingly difficult to follow in real-time. At the working level, network operation has become more important than network security.

Available test tools that support information assurance are mostly reactive in nature in that they are intended to monitor a security-related trigger for attacks or disruptions. These tools can be host based or network based. Other proactive type test and monitor tools are available that will either directly simulate an attack on known network vulnerabilities, or will continuously monitor for configuration changes to a particular network. In nearly all cases, these tools involve trained human intervention, are comprehensive to some degree, take time to run, and can generate large data files.

The primary problem with current information assurance approaches is details. Getting the knowledge needed to the right individual in order to properly carry out the required range of protection activities when attacks or even initial vulnerabilities are first detected is an issue, particularly if you don't have an individual available. Breaking this problem down further, one could conclude that a two-tier approach is needed. First, handle as many activities as possible using "smart" automated approaches. Second, when trigger indicators from local processing warrant more in-depth analysis, then perform the detailed analysis using available capabilities.

Acquisition	***knowledge needed***
Routing	***right individual***
Responsive	***carry out the required range…***
Timely	***when attacks or even initial vulnerabilities are first detected***
Analysis	***detailed analysis using available capabilities***

The generation of large amounts of raw data and how to interpret what is collected is the key for in-depth analysis. Local processing by a tool with significant security analysis capabilities can handle detailed work. Additional needs are the ability to signal detection triggers to other receptors or frameworks where this information is critical. The extraction of information from large raw data sources is also useful. Data mining is an emerging means of finding intelligence when too much information exists. When an intelligent decision is needed from a real person, data mining allows automated acquisition, generation and exploitation of knowledge from large volumes of heterogeneous information.

The ultimate goal for maintaining a real world security posture would be to turn an automated tool loose to find and report the related information or patterns needed as close to real time as possible. Other "smart" tools could then be used to gain additional knowledge about the environment or about some specific security related condition, or even correct certain conditions should they be detected. Also, the ability to data mine audit files and transmit the analyzed information would eventually become a significant tool in the security framework.

What Are Intelligent Agents

An intelligent agent is basically a system that can perform a task based on intelligence learned or rules provided. They are able to independently evaluate choices without and human interaction. When several agents are put together, they form an agency, capable of a combined range of actions. The agency picks the right agent needed to perform a specific task, using the network itself to do the processing. In other words, the agent represents the user to select and complete a required task through ruled-based criteria while using and interacting with other programs and data.

Agents are a natural extension for complementing and evolving technologies. In the future as these other technologies evolve, so to will the corresponding capability of associated agents to deliver these technologies. Some supporting and complementing technologies include:

- Object-Oriented Programming
- Neural Networks
- Fuzzy Logic
- Genetic Algorithms

Agents consist of a common architecture such as shown in Figure 1. The knowledge base contains the knowledge that has been generated as well as rules that are being followed. Libraries contain information the agent has identified. Application objects are the resources available to the agent (test tools in our case). The adapter serves as the standardized interface for the tools. The views are basically who or what the agent is capable of delivering to its user.

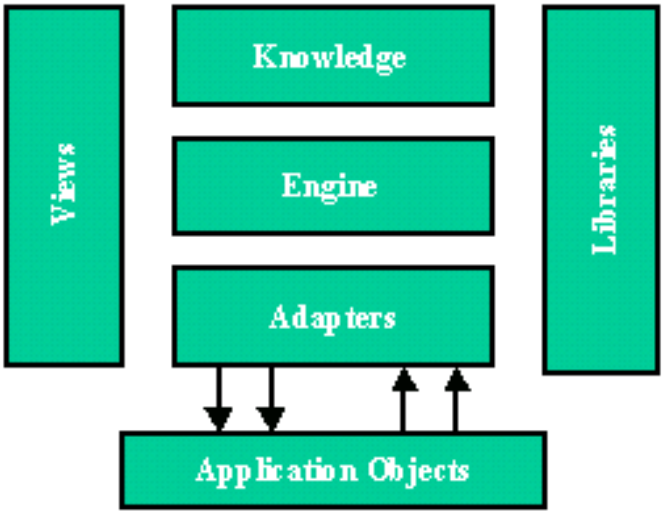

Figure 1 - Common IA Architecture

Intelligent Agent Applications

Intelligent agent based Host Security Engineering Tools are aides to the system integrator, helping to ensure that integrated systems are security hardened more consistently. CIST's approach to supervise and control information assurance tools when employed to protect the enterprise wide environment involves the use of a fast snapshot configuration management intelligent agent for the initial discovery and evaluation of security related change.

Fast agents are those that have a single task. To be effective in a large, ever changing environment, the primary change discovery agent needs to be both fast and focused. It should interpret (analyze) the initial conditions and then report specific findings to supporting agents and tools. It should also maintain raw and processed data should it be required by other agents or observers. Supporting agents and tools can be a mix and match of detectors, reactors, investigators, and controllers. Control agents can be used to manage agent interactions for many problems without the need of human intervention. When major problems require direct intervention, observers can then able to direct more comprehensive investigations or responses in a focused manner through the use of frameworks.

Frameworks are used to provide various ways for clients and servers to communicate. They could themselves be agent suites or simply a controlling agent so as not attempt to concentrate analysis and interpretation capabilities at a central location. This "centralized approach" to security management has often led to "data overload," and has been the downfall of many well-intentioned programs. Instead, the framework agents should support other agents with common services such as pre-analyzed data reporting and/or collection, information storage, event management, user interface, and task automation to organize and integrate individual probes, monitors, and sensors in the enterprise-computing environment. The framework agents should also provide the application programming interfaces and services necessary to partition and distribute applications.

With good communications, agent suites can integrate Commercial-Off-the-Shelf (COTS) and Government-Off-the-Shelf (GOTS) applications for vulnerability assessment, audit monitoring, intrusion detection, and malicious code detection and eradication and provide the following security services:

- Detect, report, and correct changes to the trusted information base
- Detect, report, and respond to intrusions, misuse, and anomalies
- Detect and eradicate malicious code
- Maintain security policies and allow strong configuration management
- Actively and passively test host and network security
- Store, sort, associate, aggregate, report and store trigger event information
- Allow a process to consolidate and respond to additional in-depth data requests
- Provide other sensor (tool) management

The fundamental difference between newer agent oriented frameworks and earlier frameworks is that an efficient framework should leave the analysis to others and simply ensure that the results are provided to the proper location, be it a person or other agent. Should an observer require further information or even raw data, the framework agent should be able to find the requested information and then move it as requested.

The Security Agent Solution

Any security agent that can solve the information assurance problem set will have needs based on what job it is doing. The primary needs of a security agent relate to information, reaction and control:

- Information needs
  Threshold determinations, chain of command, IW threat condition, extent of problem /operational condition/course of reporting action in response to an identified threat (immediate vs. controlled), control decisions, local/remote react tools initiation capability.
- React needs
  Tied to the lowest level in order to provide immediate and direct control.
  The agent must be in a position to quickly detect or re-detect subsequent vulnerability information, and react with corrective responses or send requests to a higher level authority.
- Control needs
  Standard operating procedures, firewall information, a network map (routers, hubs, etc.), and threat assessments.

Reaction and interpretation represent the most difficult challenges. For the security agent to work, it needs some rules to follow depending on where in the infrastructure the agent will operate and what we want it to do. We want the highest level agents to interpret only what is going on at their own level or at levels below them so they can follow reaction rules that are appropriate for their level. Without this level of abstraction, any agent could be overloaded with

too much information. Controls must therefore be in place that can support the reaction rules that have been decided on at the proper decision level.

Another related problem is that the distributed, hierarchical control/reporting strategy reflects that an element's authority to directly command/control the examination, monitoring and analysis of (lower-level) network assets may diminish with its degree of remoteness in the organizational hierarchy. Therefore, achieving this objective would seem to imply that a multi-threaded approach might be necessary, one level which would manage assets directly, one intended to evaluate the situational responses from local assets, and one intended to allow hierarchical approaches for management and analysis.

Intelligent agents are a natural approach to solving this problem using their remote programming capability. A common agency design might be developed that can change its structure depending on where it is located and who it needs to talk to in the hierarchy. Using an agent's remote programming (RP) capability for computer-to-computer communications, the agent approach is particularly suited to help filter and take automatic actions at higher levels of abstraction, in addition to detecting and reacting to local patterns in system behavior. The agent framework can include a distributed knowledge base which acts as a repository for reference knowledge on vulnerabilities, attack techniques/signatures, countermeasures, as well as information collected concerning detected intrusions, vulnerabilities, and corrupted software/information.

Detecting Triggers

Any network characteristics that could indicate a security problem might exist are called triggering events. A known modification to any portion of the system mission, environment, or architecture that affects the system's overall security posture would be considered a triggering event. The trigger events could include:

- The presence of a new system or connected systems
- A significantly changed system configuration
- Enabling previously disabled services
- Activation of a known vulnerable port
- Trouble/Vulnerability Test Reports
- A new security advisory indicating a problem

Note that while some of the above trigger events obviously indicate a potential security problem requiring immediate action, others might simply indicate that a network change has occurred. Testing experience provides a good feel for the types and severity of changes that may cause a vulnerability problem. Part of the knowledge a security agent should possess is the ability to first identify the trigger, and then the ability to react to the trigger in a way consistent with the threat potential.

Agency-Based Framework Objectives

The following design objectives should provide guidance in developing a realistically useable security agency-based framework.

- Develop a security specific rule base driven control agent and supporting library.
  - The agent will interpret policy and response/direct commands initially for its lowest level of planned deployment.
  - Include the capability for manual controls to implement react policy/process.
  - The agent will have the capability to recognize its location and hierarchical level.

- Incorporate into the agency framework the ability to interpret/correlate/respond to multiple information sources.
  - The agency's decision process will include an interpreter that provides the ability to direct multiple reactions/responses from various for physical/logical interfaces.
  - An initial data mining capability is necessary to identify and analyze raw archive data specific to an identified incidence. The output of the analysis should be encrypted and packaged in a small footprint to reduce storage resources and then visualized in a way that is easy to understand.
  - Should the process require additional information, the ability to mine additional non-predefined heterogeneous information sources for additional intelligence would be an added benefit.

- Incorporate the capability for the agency to remotely program other agencies.
  - Control mechanisms should be considered in the interpreter agent that would allow secure (positive) identification of users plus would provide configuration/reconfiguration information (i.e., higher command level control) for eventual higher level operation.

Information Needs – A Working Agency Suite

SAIC has been developing an agency-based framework capable of discovery, detection, protection, enforcement and reaction. The intent of the program is to provide a near-real time autonomous security capability for those faced with limited resources and increasing threats. To achieve near-real time, the detection oriented agents continuously attempt to identify anomalies by comparisons to "acceptable" conditions, and then report immediately when differences are detected. Close configuration management is the natural extension of this approach. The descriptions that follow relate to SAIC's evolving security program, and to the existing DISA and other tools that contribute to the framework.

The Trover Agent

Trover is a security specific rule base driven agent with supporting library. It has the ability to communicate with other agents. Trover, depicted in Figure 2, is effective as a quick audit and compliance validation tool that offers significant additional benefits to those organizations with large network security audit needs. It can be used as a network/security tool that discovers

information indicators including the location and description of all listening TCP ports on a (remote) host. Trover profiles the initial baseline and then sequentially scans and compares to detect potentially vulnerable changes to the initial baseline. Results provide both the initial baseline snapshot, and the history of configuration change snapshots.

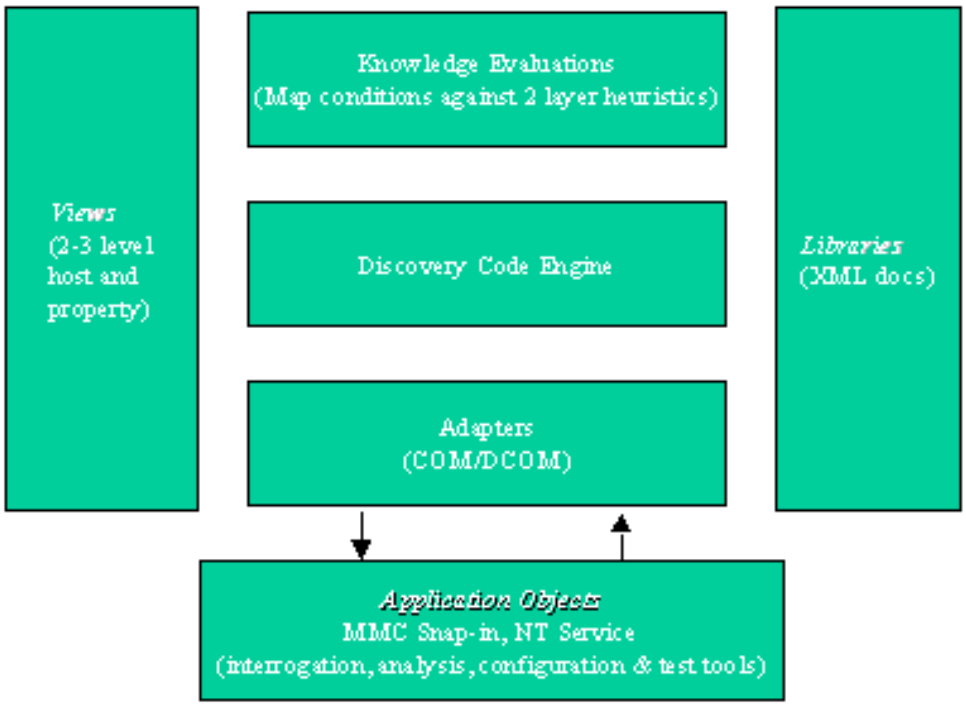

Figure 2 - Trover Components

Trover currently uses the standard Microsoft Management Console (MMC) snap-in user interface for initial set-up to determine its network boundary conditions. Once conditions are established, the agent runs in the background performing discovery functions and updating its knowledge base. When an anomaly is identified, Trovor's MMC based interface identifies the particular system involved and allows for more detailed data review or additional inquiries to take place. Trover has the interfacing structure to allow it to communicate quickly using snmp with other agents. Actual generated results are kept simple using XML documents.

Data Mining Agents

When an agency-based framework is implemented, local data processing and raw data storage can be achieved. However, at some point a need will exist to remotely analyze a particular situation to a greater degree. Figure 3 represents the agent structure for the acquisition of knowledge from wherever it may be located. Data can exist in many forms, each of which requires a special agent or tool designed to find, acquire, interpret and extract the related data into a useable and common format. Data mining agents perform this function.

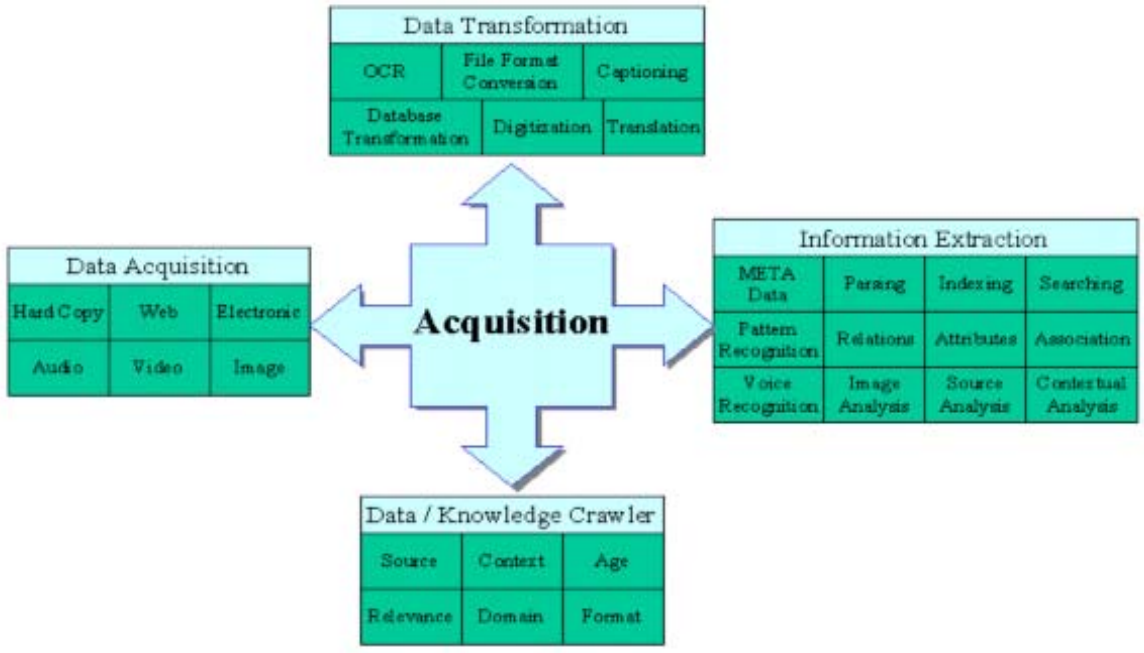

Figure 3 - Knowledge Acquisition Structure

A general, usable agent that allows remote data analysis and/or collection to all data must meet certain criteria: It must offer a single point of contact that provides uniform access to all the information available on a particular network (such as audit logs) or on the Web. It must help the user locate either pre-analyzed relevant information, or raw data if necessary. In order to do this, the system should provide recommendations for refining user queries and help the user manage and understand the complexity of the information space. It must have reasonable performance in processing user queries. It must address issues of scalability in terms of storage requirements at any of the system's distributed components. It must address scalability issues in terms of network communications, by efficiently and selectively accessing the large and rapidly growing number of information servers. Finally, it must provide for a secure means of information transfer.

Reaction/Enforcement Needs - NTSECWIZ Agent

The agent configured NT Security Wizard (NTSecWiz) supports expert security analysts in engineering a secure Windows NT platform and also system administrators in the maintenance of that secured configuration. The NT Security Wizard Agent is a supplement to the NTSecWiz tool. It allows automated remote policy enforcement as a result of agent communicated dynamic responses. The NTSecWiz performs in-depth security policy conformance analysis and object analysis.

The administrative functions of the NTSecWiz include the ability to select and apply a policy to the local or remote computer with an installed NT Security Wizard service. Alternately, the NTSecWiz allows a computer to be analyzed against a user-selected policy. The user has the option of

performing a strict or flexible analysis with the flexible analysis applying a number of heuristics designed specifically to reduce false positive results. The result of the analysis, in either case, is a report of exceptions and configuration recommendations.

NTSecWiz is based upon a modular model. The security configuration information (profile) is collected from a target machine. Both the application itself and the service are capable of doing this. This profile can be translated into a readable report. Alternately, the "profile" can be used by the analysis engine to perform comprehensive analytical functions, such as security policy analysis. The analysis engine also includes the ability to translate a policy into configuration instructions. The configuration engine utilizes these instructions to reconfigure a local or remote computer as appropriate.

PConfig and the UNIX Utility Agent (UUA)

SPConfig stands for "Security Policy Configuration". SPConfig currently exists as a Government owned manual security policy comparison and enforcement tool for DII COE Solaris configured servers. SAIC has modified to tool to also work with IRIX and LINUX platforms. The modified tool is now called PConfig. The *pconfig* Perl script functions as the controller module for program operation. It is also the command-line user interface. Important preliminary checks, such as command-line argument processing, UNIX Utility presence checks, and verification that the program operator is root are performed. The script will launch either compliance mode or configuration mode operations.

UUA is an intelligent agent that uses PConfig and provides a capability similar to NTSecWiz policy management for UNIX based systems. Running as a background agent on the host, it allows secure automated remote policy enforcement as a result of agent communicated dynamic responses. Just as with NTSecWiz, UUA can perform in-depth security policy conformance analysis and object analysis.

Coordination and Control Needs - Situation Response Agent (SRA)

SRA in based on a tool called Nmap. It provides programmed multi-agent coordinated responses at the local level to agent discovered events when running in its near real-time reaction mode. Using encrypted protocol it can provide near real-time manual or automated response commands to configuration or test agents based on inputs received from discovery agents.

An additional feature of SRA is its projected ability to provide CJCSI 6510.01B required incident reports on request. Since SRA can be configured to support hierarical lines of authority, it can communicate with other discovery and in-depth analysis agents, or with centralized databases, to obtain and deliver raw or pre-analyzed data from these agents to observers at various command levels.

Internal Situation Awareness Concentrator (ISAC)

ISAC is not an agent but a data-normalizing program. It has the ability to work with other internal security agents to normalize their information into a common CIDF format. It does this

using secure snmp protocol and a small footprint database. The pre-analyzed data or trigger indicator information provided by various sensor tools and agents, including Trover, NTSecWiz, UUA, Computer Misuse and Detection System (CMDS); etc. is placed on the database in flat files for later extraction.

Developing an Advanced Agency for React and Detect

The ultimate objective of a near real-time network security agency would be to have the agent act independently, based on pre-defined or even newly discovered conditions, to react and protect against a discovered threat. Future research will be focused on enhancing the interfacing and control capabilities for both automated network configuration and test tools, and well as enhancing the analyzing and reporting capabilities of various detection agents. Another significant effort will involve enhancing the ability to data mine, concentrate and pre-analyze collected raw data across the enterprise in order to reduce the amount of information transmitted to higher levels of authority. The intent of this effort will be to develop the independent detect and react security agency intended for broad application to hierarical large-scale networks.

Conclusion

Agent-based frameworks that support large-scale networks are achievable. Near real-time detection and reaction using “snapshots” and quick comparisons is also achievable. The current problem with any higher level agent is a common and robust protocol that each other agent or agency can use to securely communicate with all other agents in a manner that facilitates quick responses. This quick response capability for ever larger network conditions, when coupled with heuristic data mining capabilities and enhanced discovery capabilities, enables the agency approach to provide a good initial solution to the existing real world problems associated with information security. It also provides the foundation for further evolution based on future technology advances.