# On Ubiquitous Network Security and Anomaly Detection[*]

Colin Van Dyke Çetin K. Koç
Electrical & Computer Engineering
Oregon State University
*{vandyke,koc}@ece.orst.edu*

**Abstract**

As networking trends move toward ubiquitous structuring schemes, the problem of security has taken on an increasingly important role. As a result, we must look to new security paradigms that address the new problems associated with these networks. This paper is a summary of research in progress on the *Wintermute Project* at Oregon State University, which was created to investigate security for ubiquitous networking schemes.

## 1 Introduction

As we move into an era of computing trends aimed at realization of a ubiquitous and pervasive Internet, a constant limitation on success and growth is security. Existing technologies for protecting communications have come under constant scrutiny for their weaknesses against eavesdropping and danger of providing unwarranted access to private networks [4]. Furthermore, in some instances this unwarranted access can allow a malicious party to perform changes to the infrastructure allowing for larger-scale attacks. This ubiquitous computing paradigm has created new challenges in security and requires that we develop new approaches to address both existing and new security problems. As a result, much of our research has focused on creating a platform to address the security needs of dynamically changing network architectures. Through this work we have been able to assimilate existing solutions to network security in order to address new security problems, as well as develop new ways of combating growing security needs. One such method is using a scheme of *Network Anomaly Detection* in order to identify and address infrastructure changes across multiple nodes of a network that could perhaps signal some distributed attack or malicious entity [1]. Below we describe our research in progress to develop a scalable and robust security architecture targeted at ubiquitous networking schemes.

## 2 A Security Architecture for Ubiquitous Networking Schemes

In developing this architecture we sought to utilize emerging technologies with existing techniques to arrive at a solution that provides a high level of transparent security for network users. The catalyst for this is a base *agent architecture* that includes security enhancements and enforcements allowing for creation of a trusted computing base within a given local network. One way to envision this is as a "Circle of Wagons" in which the outer perimeter of the network is the most vulnerable and likely to be attacked. Only trusted nodes are allowed to form this perimeter and are

[*] *2003 Symposium on Applications and Internet - Workshop on Security and Assurance in Ad hoc Networks,* pages 374-378, Orlando, Florida, January 27-31, 2003.

therefore subjected to a thorough security analysis before network access is granted. A further description of this agent architecture follows.

## 2.1 The Base Security Layer

The lowest level of our architecture is an enhanced agent platform that functionally allows for the dissemination and gathering of information on a network-wide basis. This provides the ability for monitoring of security on individual and multiple nodes, as well as integrity throughout the network topology. The agent platform is based on the Java™ programming language due to portability across various architectures and built in access control policy. Additional enhancements have been applied to the agent platform to ensure that it does not introduce any additional vulnerability into the network.

Attacks against an agent by its corresponding host [2] are believed to be non-existent as the structure of our agent platform creates a trusted computing base, where each node on the network is treated as a trusted entity. This may be addressed differently in the future to further enhance the security of our network against malicious entities.

## 2.2 Overview of Agent Types

Our security architecture relies on the combined work of three main agent types: Nodal, Manager and Root Agents. These 'classes' of agents work together much like a hive of bees, where each individual has their own contribution to the greater purpose. An overview of the specific agents and their related jobs follow:

Nodal Agent:

The *Nodal Agent* is the lowest level agent in our architecture and as a result the most prevalent. By this we mean that it functions at no abstraction, and is specifically designated to an individual network node. All nodes on the network have a corresponding Nodal Agent. The primary job for a Nodal Agent is to monitor local security. In the event of a security breach or attack, the Nodal Agent must first notify a Manager Agent and then perform functions to try and counter said attack. If that is unsuccessful, the Agent must remove the node from the network to prevent further exacerbation of the attack. In addition to monitoring local security, a Nodal Agent is in charge of screening a local network node for security vulnerabilities and applying the corresponding patches. This must be performed before the node is allowed access to network resources. If necessary (in the case that the node is requesting access to sensitive network resources) the Nodal Agent will perform some form of authentication with the user.

Manager Agent:

The *Manager Agent* acts as a middle party in the overall hierarchy of our security architecture. Its primary goal is to monitor the state of its corresponding network infrastructure and Nodal Agents. If an attack is triggered, Manager Agents are responsible for determining if it is distributed in nature, and taking the corresponding measures to reroute network traffic and minimize damage. They may also spawn another version of themselves to aid the Nodal Agents in combating the attack. The information 'known' by each Manager Agent must remain synchronous with all other Manager Agents.

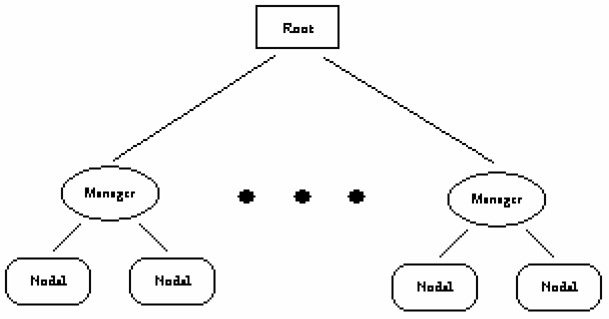
Therefore, some measure of synchronization capability must be included in the final design of the Manager Agent. All Manager Agents must report information to and take orders from a Root Agent if need be.

Root Agent:

The *Root Agent* exists as a distributed entity within the architecture; that is, multiple forms reside only on trusted servers and remain synchronous with each other throughout operation. No root agent presides over another. The Root Agent's sole purpose is to monitor the actions of all other agents and, if necessary, assign specific tasks to individual Manager and Nodal Agents. Essentially, it acts as an administrator on a network and is most often controlled by a human counterpart. Therefore, usage of the Root Agent is subject to strong authentication. Thorough security provisions are applied to the Root Agent, as it is the controlling entity within our security architecture. Due to its distributed nature, successful attack and removal of a single Root Agent does not necessarily affect the function of the network.

## 2.3 Network Topology

We now wish to view the structure of our security platform at a network level so that we can further understand how it applies to ubiquitous networking schemes. The goal of our platform is to view the network at an abstract level in order to minimize the amount of concrete structure that does not always exist in modern networks. This allows us to function in an environment where nodes are constantly added and removed from a network. No nodes on a network are treated as static with the exception of one group, the *Root* nodes. These trusted nodes serve as an overseer to all other nodes and interconnections. The entire network is focused around this group of nodes, though communication provided by the agent platform does not necessarily pass through them. On the contrary, it allows for monitoring and modification of network topology in case of attack. As one can stipulate, the only agents operating with the group of Root Nodes are Root Agents. However, the Root Agents are often treated as one entity, as their operation and modification must remain synchronous with the others. From this point, an administrator can add agents to the network as well as assign specific tasks to various Manager Agents. As one can imagine, Root Nodes can easily become the focal point for attack, and therefore must be subjected to the highest level of security. In the case of a successful attack against a Root Node, Manager Agents must be able to either re-associate with another Root Node or terminate their execution as well as disable (temporarily) their corresponding network node. The general topology of the network exists as follows:



Manager Agents generally reside on pseudo-static network nodes, such as agent-enabled routers and servers. If need be (as in the case of an attack) they can migrate between any network node to perform the necessary tasks as well as re-associate with other superior agents. In addition, they can trigger other Manager and Node Agents to temporarily disable the functionality of their corresponding network nodes, reroute network traffic around affected network areas, and subsequently destroy all compromised agent data.

Nodal Agents reside on the outlying and commonly dynamic elements of a network. This includes all machines both statically connected to the network but not permanently on, as well as those machines that may travel in and out of network connectivity (such as those on a wireless network). Each time a machine requests access to network resources, it must first go through a rigorous security analysis and authentication against a database of previous access attempts. This is all performed transparent to the user and is handled entirely by the Nodal Agent. The job of the agent after access is granted is to monitor local security and communicate any concerns to a higher authority (such as Manager Agents or the Root Agent). An efficient method to accomplish this is subject to future work.

# 3 Security Provisions

The primary goal of this architecture is to provide transparent security in a uniform manner on a network that may or may not be ubiquitous in nature. The design was intended to apply to ubiquitous networks as they become more prevalent, but is not necessarily limited to such networks. Though our architecture provides a high level of security on a number of different levels, we would like to focus on three main concepts that make it unique.

## 3.1 Trusted Computing Base

Many networks currently deployed exist as a trusted computing base, though there is no basis for ensuring and validating the level of trust. As a result, any node on a network can either function as a malicious entity unchecked, or act as a likely target for intruders to use as a gateway to the remainder of the unprotected internal network. Our architecture seeks to remedy this through enforcing a strict security policy on each network node. As stated previously, each node must undergo a thorough security analysis and patching before access is granted to network resources. In doing this, we ensure that a uniform secure base exists on the network, and therefore using an individual node to gain access to greater resources is restricted. In addition, we attempt to make it difficult if not impossible for the user of an agent type to modify or access nodes containing resources outside of the corresponding agent domain (Nodal, Manager, Root).

## 3.2 Authenticated Access

Authentication allows us to grant access to certain trusted users as they move across different machines in a network. Therefore, no user is bound to a specific machine. As some users have greater access rights to network resources when compared to a “default” user, they are allowed to authenticate with the local agent that then makes the access request throughout the agent hierarchy. This can also be extended to apply to roaming profiles throughout a network. A user can use one machine (such as a handheld device) to enable access rights on another machine given that the two machines are directly connected. In doing this, they can effectively use any node on a network as if it were their own node in the gauge of access rights.

## 3.3 Network Anomaly Detection

*Network Anomaly Detection* is a security method we have been developing [1]. It is an abstraction of existing Intrusion Detection techniques to the network level allowing us to simultaneously monitor the security of multiple nodes as well as the network infrastructure. It searches for anomalies within the network structure that could be possible indications of distributed attack and perhaps takes action against such an attack. This is somewhat similar to the concept of network-based Intrusion Detection Systems, but differs on several key points. The need for this scheme arises from the current computing trends toward the network computer. We see distributed computing and environments gaining ground as the major technological trend of today, and Network Anomaly Detection addresses the adverse security needs of this new paradigm. Utilizing this technology, we create an architecture that allows for broad security throughout a ubiquitous networking scheme. For more information on Intrusion Detection Systems the reader is referred to [3] and [4].

## 4 Future Work

The most fundamental element of future work is taking the agent platform developed by TU Berlin [6] and adding security enhancements making its use viable in a possibly malicious environment. In addition, we need to focus on creating an effective security policy and enforcement measures to be included in our security architecture. An interface must be created for work with existing Intrusion Detection programs at the nodal level. We also need to analyze techniques for efficient synchronization of information between Management Agents on the network. For testing purposes, a prototype for efficient Network Anomaly Detection across wide-scale ubiquitous networks needs to be developed. The development work continues, and we will produce a preliminary report on efficiency and security issues in the near future.

## Bibliography

[1] C. Van Dyke and Ç. K. Koç. Network Anomaly Detection. In Progress.

[2] T. Sander and C. F. Tschudin, Protecting mobile agents against malicious hosts. *Mobile Agent Security*, pages 44-60, LNCS 1419, Springer Verlag, 1997. http://citeseer.nj.nec.com/sander98protecting.html

[3] J. P. Anderson. Computer Security Threat Monitoring and Surveillance. Technical Report, James P. Anderson Co., April 1980.

[4] D. E. Denning. An intrusion detection model. *IEEE Transactions on Software Engineering*, SE-13(2):222-232, February 1987.

[5] Nikita Borisov, Ian Goldberg, and David Wagner. Intercepting mobile communications: The insecurity of 802.11. In *Proceedings of MOBICOM 2001*, 2001. http://citeseer.nj.nec.com/article/borisov01intercepting.html

[6] R, Sesseler and S. Albayrak. JIACIV – An open, scalable agent architecture for telecommunications applications. *First International NAISO Congress on Autonomous Intelligent Systems, ICAIS.* To be published 2002.