A Framework for Cooperative Intrusion Detection

Presented By:Lo’ai hattar
Supervised By: Dr. Lo’ai Tawalbeh

New York Institute of Technology (NYIT)-Amman

Introduction

- *Experienced intruders often launch multiple simultaneous sessions to attack a system or set of systems.*
- *Information sharing is necessary if certain attacks are to be detected .*
- *Although traditional crime prevention and law enforcement agencies have relied on cooperative techniques for years .*
- *no true automated cooperative systems have been developed to address intrusion detection needs.*
- *One reason for this is that no mechanisms exist for sites without central administration to safely and conveniently share data that may be used to track widespread intrusions*
- *This paper proposes principles for a framework to support cooperative intrusion detection across multiple domains*

Principles for a cooperative intrusion detection system framework

- *information sharing takes place both between distinct domains and within domains*
- *Domains may have varying or even conflicting policies, and will not necessarily have the same mechanisms in place to identify system misuse*
- *it useful to consider these principles with respect to the relation between participant pairs, and have identified*

the following primary relationships:

Peer:

- *a relationship of equals between hosts, typically in different policy domains.*
- *Neither host controls the other, although they may send requests or information between themselves.*
- *Peers do not necessarily trust one another, and the level of trust is not necessarily identical.*

Manager:

- *a manager is a host that provides instructions regarding which data is to be collected .*
- *Managers set a central policy for a group of hosts, generally within the same policy domain.*
- *Managers need not trust their subordinate hosts.*

Subordinate/Managed Host:

- *a host that receives some or all of its data collection and transmission policy from outside.*
- *Managed hosts may modify or add to this policy, and may themselves manage other hosts.*
- *Managed hosts must fully trust their managers, and will usually be within the same policy domain.*

Slave Host:

- *a host that receives all of its data collection and transmission policy from outside.*
- *Slave hosts may not modify or add to this policy, although they may themselves manage other hosts.*
- *Slave hosts must fully trust their managers, and will always be within the same policy domain.*

Friend:

- *a relationship of equals between hosts. Neither host controls the other, although*
- *they may send requests or information between themselves.*
- *Friends always trust one another,and the level of trust is identical. Friends should be within the same domain.*

Symbiote:

- *a relationship of interdependent hosts. Neither host controls the other, although*
- *they may send requests or information between themselves.*
- *Hosts with this relationship are expected to be `identical' in terms of policies and security labels.*

principles important in the development of a framework for cooperation between domains:

Local Policy Control

- *is one of the most important facts of cooperative intrusion detection systems.*
- *Cooperating domains may not fully trust one another and will be highly unlikely to grant any outsider the right to change their internal methods of misuse data collection or information flow.*
- *The local host should always determine whether a given interaction should take place based on its own local policies.*
- *Even for hierarchical relationships such as master/slave, the slaved host should have facilities for checking its own policies before reacting to the incoming message.*

Autonomous, but cooperative data collection:

- *Although the data collected and shared is determined locally, hosts and domains should share relevant information.*
- *This may involve collecting and sharing data which is irrelevant to the source host's own security policy but is needed by a peer or manager.*
- *However, even when hosts are obtaining data needed to identify policy violations for external domains, the decision regarding whether to collect and transmit this data is local, and should include the realization that the recipient will `own' the data once transmitted and may decide to disseminate it further.*

Data reliability:

- *Actions based on data obtained elsewhere should include a local trust factor, since incoming data may not be reliable.*
- *For example, the source host may have been compromised and be unknowingly transmitting misleading data.*
- *Thus, the intrusion detection system using data obtained via the cooperative framework should employ a method which takes into account the local host's trust in the authenticity and the integrity of the data.*

Policy enforcement and identification:

- *The enforcement of policy and the identification of policy violations should be separate issues.*
- *Hosts may be identifying policy violations for another domain than their own, but they should not be responsible for enforcing those policies.*

Validated Transactions:

- *It will be necessary to perform authentication of some kind between cooperating hosts.*
- *This authentication might involve digitally signed messages, for instance.*

Structure of sharing:

- *Information sharing may be both horizontal and vertical.*
- *For example, a manager and its subordinates may perform vertical sharing, where subordinates transmit data \upwards" to the manager and managers transmit commands \downwards."*
- *The trust relationships between participants are likely to be strong in the downward direction (subordinates trust their manager) and weaker in the opposite direction (managers may not fully trust their subordinates).*
- *Between peers, sharing should be horizontal, due to the more collaborative nature of their relationship.*

Data collectors:

- *Data collectors should be overlapping and provide both data reduction and sanitization. Overlapping data collectors are needed whenever one data collector might be subverted or become unavailable.*
- *Data reduction is important to reduce the extraneous data transmitted between sharing partners.*
- *Data sanitization is needed to eliminate host and network specific attributes of the data. This is important since such data might cause a security risk to the transmitting host.*

Integrated audit tool management and visualization systems:

- *As data collection systems become more complex and handle more and more systems, manual configuration of audit tools becomes impossible.*
- *Further, examination of voluminous textual information for potential violations is difficult; humans are far better at processing and identifying oddities in graphical forms while systems produce faster initial response.*
- *Thus, any data management system should include methods for managing sets of audit tools and providing graphical views of examining such data.*

Policy Sharing Issues

- *We divide policy into three components:*
  - *access control*
  - *integrity*
  - *cooperation*

- *each host should let its local policy and local ratings govern its interactions*
- *This permits domains to cooperate even when they disagree on some policy issues*
- *the transmitting host relies on its cooperation and access control policies to determine which data it will send.*
- *the receiving host relies on its integrity policy to determine how(and whether) the data will be used*

The access control policy

- *is used to determine whether a subject (in the context of the evaluation domain) has sufficient clearance to perform a given operation on a particular object.*

The integrity policy

- *is used to determine whether information should be allowed to be modified or added to a system*

The cooperation policy

- *determines whether data which the access control policy permits to be shared will in fact be shared.*
- *The cooperation policy is somewhat reminiscent of a discretionary access policy, since decisions to cooperate are based on the relationship between hosts and should be made on an individual basis.*
- *However, cooperation requires expenditure of resources to collect, store, and transmit data, and to manage incoming requests, and sharing of data opens the risk of releasing data which may increase a participant's own vulnerability*
- *the cooperation policy should be developed separately*

Data Filtering

- *Filtering is needed for*
  - *data reduction and*
  - *to protect sensitive internal resources.*
  - *useful mechanism for enforcing aspects of the access control policy as well as an important performance measure.*
- *Filtering includes both :*
  - *data reduction*
  - *data sanitization.*

data reduction

- *benefits the `requesting' host, since it reduces the transmission of extraneous information. Although the source host*
- *must perform additional data processing, communication costs should be reduced for it as*
- *well.*

Data sanitization

- *benefits the source host by removing potentially sensitive information which might induce or expose a vulnerability.*

Filtering

- *may be performed during*
  - *Transmission: Data transmission filtering is done with respect to the specific target, can be host specific, and has a fine granularity*
  - *data collection,*
  - *and/or storage.*
- *Collection/storage filtering is done with respect to a general policy regarding what will be stored, is less specific, and will tend to result in*
- *either a generally lower level of sensitivity of data being transmitted (since the data will be highly sanitized)*
- *or a reduced amount of information being transmitted (since fewer hosts will be allowed to read the data).*

Hummingbird System

- *The HMMR protocol has been designed*
  - *To address the requirements needed to permit network sites to share security relevant data while still retaining local control of data gathering activities,*
  - *Deciding locally how much trust to place in data received from outside sites, and*
  - *Determining how much data to share with outside sites.*

- *This system is being used as a test bed for exploring practical issues in sharing data between sites, such as how much reliance to place on offsite intruder alerts and misuse data,*
- *how data may be shared safely, which collected data makes a quantifiable contribution to intruder identification, and so on.*
- *Hummingbird is formulated as a distributed system for managing misuse data.*
- *Hummingbird employs a set of Hummer agents, each assigned to a single host or host set*

- *Each Hummer interacts with others in the system through manager, subordinate, and peer relationships. Managers may transmit commands to subordinates; such commands include requirements to gather/stop gathering data, to forward/stop forwarding data, and the like.*
- *Peers may send requests for data forwarding/gathering/receiving to peers; the peer decides whether to honor such requests*
- *Subordinates may send requests to their managers as well.*

- *Kerberos is used for authentication and encryption of communication between hummers and for authentication between Hummers and their associated database.*
- *Individual Hummers share data based on their own locally controlled and set policies regarding trust, information flow, and cooperation. Simple filtering is also performed by each host.*

- *The prototype Hummers use POSTGRES, an SQL-compliant database, for storing misuse data. By using an SQL database, it is possible to easily obtain subsets of data to:*
  - *Generate reports,*
  - *generate information for an intrusion detection system,*
  - *and data for the visualization system*
- *and supply data to the feature selection routines (under development).*
- *The POSTGRES database in the prototype system can supply this information either in text format or as an HTML table.*

Audit tools

- *Audit tools run either individually under each Hummer's control or else as a suite through the Audit Tool Manager .*
- *These audit tools collect security relevant data from system log files, by running system utilities, or by running specialized auditing processes.*
- *Each Hummer's configuration includes information about the audit tools it is running, so that if a Hummer is stopped and restarted, it can regain control over the tools or start them up again if necessary*

- *Alert tools are a special case of audit tools, since they only examine a Hummer's own log files. These alert tools perform simple security assessments by looking for indications that an intrusion may be in process.*
- *Reports from the data collected by a Hummer may be obtained either directly from*
  - *the POSTGRES database*
  - *or through the visualization tool*

The audit tool manager

- *Atom, is a front end for managing Hummingbird audit tools based on the perceived level of threat to the network .*
- *It implements a graphical user interface that allows the user to manage audit tools on a network or a single host.*
- *Atom is integrated with the Hummingbird system to improve the management of audit tools and increase specialized data collection.*

- *Each audit tool must be registered with Atom.*
- *Registration information includes*
  - *the name of the audit tool,*
  - *a security classification level,*
  - *parameters of the tool,*
  - *textual description,*
  - *tool classification type,*
  - *directory where the tool is located,*
  - *and the default execution of the tool.*
- *The default execution is the command used to start the audit tool on the host.*

- *The primary purposes of Atom is to increase and ease the task of performing specialized data collection.*
- *This is accomplished by assigning a security classification level (SCL) to a suite of audit tools.*
- *The framework for a SCL is a descriptive classification level name determined by the user and assigned to one or more audit tools.*
- *Typical SCLs for a network may be none, possible intrusion, and ongoing intrusion.*
- *Using this as a scenario, the classification level none is considered to be the default, or normal level of data collection on the network.*
- *When potential security risks develop, the system administrator can switch to the possible intrusion or ongoing intrusion SCLs, increasing the data collection by reconfiguring the operating set of audit tools.*

- *One of Atom's more important capabilities is the ability to start or stop a suite of data collection tools on a host or network with the click of a single button.*
- *These suites of audit tools allow Hummingbird to be tailored to collect information important in the detection and/or prevention of specific network-based attacks, or to increase the level of auditing when an attack is thought to be underway.*
- *Suites of tools which are often used together may be developed and managed under a single name.*

- *This also permits easy \upgrading" of system auditing when the perceived level of system threat increases, and an easy way to decrease system auditing when the perceived level of threat is low.*
- *Future versions of Atom will include decision-making tools automating SCL choice if desired.*
- *In addition to tool suites, Atom provides several advantages.*
  - *First, all the tools are now managed by one central manager, making it easy to identify the security mechanisms available to perform intrusion detection or auditing.*

- *Since each tool is registered with the manager and has a tool classification type, the user can easily identify and launch tools with particular characteristics, such as login monitoring tools.*
- *The parameters and the default execution command of the audit tool are registered providing efficient information about the tool without searching the system for it.*
- *In addition, scripts that are developed and used with Hummingbird can be registered and managed by Atom.*
- *Finally and most importantly, it creates a more secure and integrated system with Hummingbird for handling audit data.*

Visualization

- *is a useful part of any intrusion detection system, since presenting information graphically to the system security officer can greatly ease the task of identifying intrusions by making use of human ability to more rapidly perform pattern matching with graphical data than with raw text*

Related Systems

- *Earlier studies indicate that detection of certain attacks required data from multiple hosts on a network, and that networks under centralized control could effectively combine data sources to minimize damage*
- *(DIDS) system addressed system attacks across a network.*
- *Attacks such as doorknob, chaining, and loop back could be detected when data from hosts within a given network was combined*

- *Doorknob attack is considered to be an attempt to break into a system by testing one or more passwords |i.e., rattling the door|on a system.*
- *Chaining refers to an intruder's attempt to hide his/her origin by logging in through a sequence of hosts,*
- *and a loopback attack is one in which an intruder combines chaining with a visit to an earlier host in the sequence (often with a different login name).*

- *Without data combination, threshold-based intrusion identification schemes are easily subverted by reducing the volume of attacks directed at a particular host to avoid detection.*
- *DIDS combined data from hosts within a network under centralized control, but clever attackers could still subvert DIDS by reducing the volume of attacks for a given network.*
- *Data sharing as proposed in this project should prove to be a useful way to detect such attacks.*

- *A smaller-scale prototype system for Hummingbird permitted different trust levels between \neighborhoods of networks", and these networks were not assumed to fall under any central jurisdiction.*
- *However, this prototype did not explicitly distinguish between policy decisions based on information flow, trust, or cooperation levels.*
- *In contrast, the EMERALD system (SRI International), is intended to address intrusions within very large networks .*

- *EMERALD also considers separate administrative domains.*
- *Although these domains are assumed to lie within a single corporate structure,*
- *EMERALD includes features for handling different levels of trust between the domains from the standpoint of a centralized system: individual monitors are deployed in a distributed fashion, but still contribute to a high-level event-analysis system.*
- *EMERALD appears to scale well to very large domains.*
- *Another network-based intrusion detection system, GrIDS, uses graphing techniques to detect coordinated network attacks .*

- *This is done by representing the data collected from hosts and networks as activity graphs.*
- *These graphs are then analyzed to detect potential violations. Graph reduction techniques are used to reduce the volume of data represented in the graphs.*
- *If these reduction techniques are successful, they should permit scaling of GrIDS to large-scale systems.*