A Mission-Impact-Based Approach to INFOSEC Alarm Correlation†

Phillip A. Porras, Martin W. Fong, and Alfonso Valdes
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025-3493
{porras, mwfong, valdes}@sdl.sri.com

Abstract

We describe a mission-impact-based approach to the analysis of security alerts produced by spatially distributed heterogeneous information security (INFOSEC) devices, such as firewalls, intrusion detection systems, authentication services, and antivirus software. The intent of this work is to deliver an automated capability to reduce the time and cost of managing multiple INFOSEC devices through a strategy of topology analysis, alert prioritization, and common attribute-based alert aggregation. Our efforts to date have led to the development of a prototype system called the Mission Impact Intrusion Report Correlation System, or M-Correlator. M-Correlator is intended to provide analysts (at all experience levels) a powerful capability to automatically fuse together and isolate those INFOSEC alerts that represent the greatest threat to the health and security of their networks.

1. INTRODUCTION

Among the most visible areas of active research in the intrusion detection community is the development of technologies to manage and interpret security-relevant alert streams produced from an ever-increasing number of INFOSEC devices. While the bulk of the work in security alert management and intrusion report correlation has spawned from the intrusion detection community, this paper takes a much broader definition of alert stream contributors. Over recent years, the growing number of security enforcement services, access logs, intrusion detection systems, authentication servers, vulnerability scanners, and various operating system and applications logs have given administrators a potential wealth of information to gain insight into security-relevant activities occurring within their systems. We broadly define these various security-relevant log producers as INFOSEC devices, and recognize them as having potential contributions to the problems of security incident detection and confidence reinforcement in discerning the credibility of INFOSEC alarms.

† Supported by DARPA through the Air Force Research Laboratory, contract number F30602-99-C-0187.

Unfortunately, this broader view of alert stream contributors adds to the complexity facing intrusion report correlation systems, illustrated in **Figure 1**. INFOSEC devices range greatly in function, even within a single technology. For example, within the intrusion detection space, the variety of analysis methods that may be employed, the spatial distribution of sensors, and their target event streams (network traffic, host audit logs, other application logs), increases the difficulty in understanding the semantics of what each sensor is reporting, as well as the complexity of determining equivalence among the intrusion reports from different sensors.

The motivation for our work is straightforward: as we continue to incorporate and distribute advanced security services into our networks, we need the ability to understand the various forms of hostile and fault-related activity that our security services observe as they help to preserve the operational requirements of our systems. Today, in the absence of significant fieldable technology for security-incident correlation, there are several challenges in providing effective security management for mission-critical network environments:

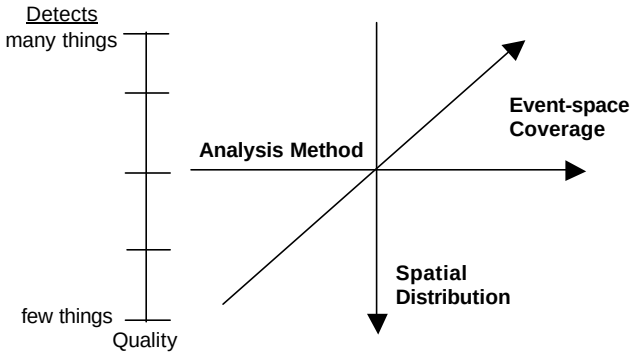

Figure 1 — Alert Stream Complexities

- Domain expertise is not widely available that can interpret and isolate high threat operations within active and visible Internet-connected networks. Also not widely available are skills needed to understand the conditions under which one may merge INFOSEC alerts from different sources (e.g., merging firewall and OS syslogs with intrusion detection reports). In an environment where thousands (or tens of thousands) of INFOSEC alarms may be produced daily, it is important to understand redundancies in alert production that can simplify alert interpretation. Equally important are algorithms for prioritizing which security incidents pose the greatest administrative threats.
- The sheer volume of INFOSEC device alerts makes security management a time-consuming and therefore expensive effort [Lev01]. There are numerous examples of organizations that have found even small deployment of IDS sensors to be an overwhelming management cost. As a result, these IDS components are often tuned down to an extremely narrow and ad hoc selection of a few detection heuristics, effectively minimizing the coverage of the IDS tool.
- In managing INFOSEC devices, it is difficult to leverage potentially complementary information produce from heterogeneous INFOSEC devices. For example, is captured in a firewall log, is typically manually analyzed in isolation from potentially relevant alert information captured by an IDS, syslog, or other INFOSEC alert source.

The remainder of this paper describes the design, implementation, and illustrative experiments developed during a two-year research study of IDS interoperability and intrusion report management that address the above issues.

2. M-CORRELATOR ALGORITHM OVERVIEW

M-Correlator is designed to consolidate and rank a stream of security incidents relative to the needs of the analyst, given the topology and operational objectives of the protected network. **Figure 2** illustrates the conceptual elements of the M-Correlator system. The following discusses these conceptual elements, as designed and prototyped in the M-Correlator processing algorithm. In subsequent sections, each step of processing is further described.

The first phase of INFOSEC alert processing involves dynamically controllable filters, which provide remote subscribers with an ability to eliminate low-interest alerts, while not preventing INFOSEC devices from producing these alerts that may be of interest to other analysts. Next, the alerts are vetted against the known topology of the target network. A *relevance score* (Section 2.2) is produced through a comparison

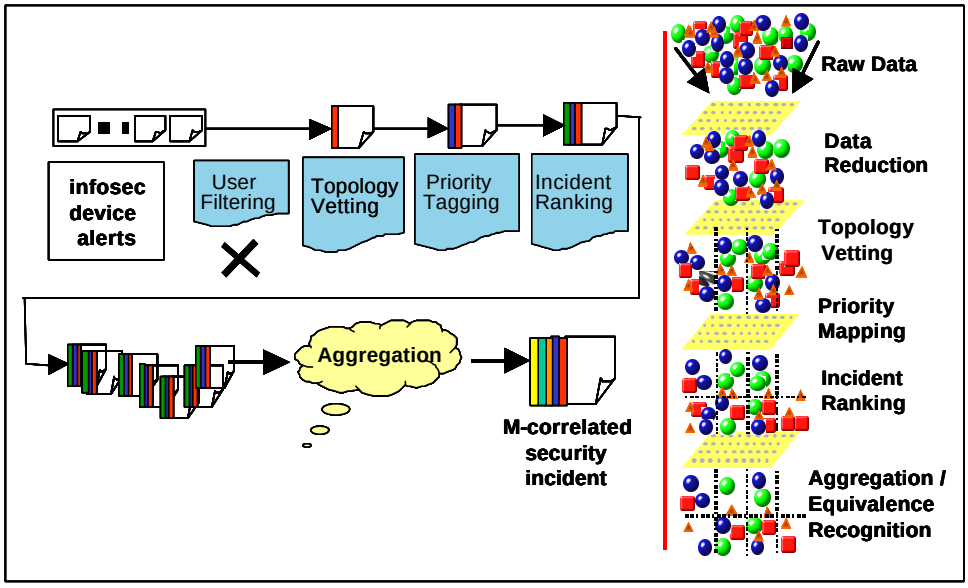

Figure 2 — Algorithm Overview

of the alert target's known topology against the known vulnerability requirements of the incident type (i.e., incident vulnerability dependencies). Vulnerability dependency information is provided to M-Correlator through an *Incident Handling Fact Base* (Section 2.1). Next, a *priority calculation* (Section 2.3) is performed per alert to indicate (a) the degree to which an alert is targeting a critical asset or resource, and (b) the amount of interest the user has registered for this class of security alert. Last, an overall *incident rank* (Section 2.4) is assigned to each alert, which provides a combined assessment of the degree to which the incident appears to impact the overall mission of the network, and the probability that the activity reported in this alert was successful.

M-Correlator next attempts to combine related alerts with an attribute-based *alert clustering algorithm* (Section 3). The resulting correlated incident stream represents a filtered, lower-volume, content rich security-incident stream, with an incident-ranking scheme that allows analysts to identify those incidents that pose the greatest risk to the currently specified mission objectives of the monitored network.

2.1 AN INCIDENT HANDLING FACT BASE

M-Correlator includes an *Incident Handling Fact Base* that provides the necessary information to optimally interpret alert content against the mission specification and relevance analysis. The incident handling fact base provides critical information needed to

- Augment terse INFOSEC device alerts with meaningful descriptive information, and associate alerts with M-Correlator-specific incident codes and classifications
- Understand the dependencies of incident types to their required OS versions, hardware platform, network services, and applications
- Understand which incident types can be merged by the M-Correlator alert clustering algorithm

Table 1 enumerates the field definitions of entries in the M-Correlator incident handling fact base. Entries in this fact base are referenced in subsequent sections, which describe topology vetting, prioritization, incident ranking, and alert clustering. The current M-Correlator fact base provides incident definitions for more than 1,000 intrusion report types from ISS's Realsecure, Snort [Roe99], the EMERALD [Por97] suite of host and network-based intrusion detection sensors, and Checkpoint's Firewall-1 product line. Incident types that are not represented in this fact base can still be managed and aggregated by the M-Correlator; however, the advanced alert clustering and relevance calculations are not performed on alerts that are absent from this fact base.

FIELD TYPE	DESCRIPTION
Incident Code	A unique code to indicate incident type. These codes have been derived from the original Boeing/NAI IDIP incident codes that were used by the Common Intrusion Detection Framework CISL specification [Kah99]. A mapping between this incident code and other well-known attack code specifications such as Bugtraq ID, CERT ID, and MITRE CVE codes is available using the References field.
COTS Codes	An equivalent code listing of well-known commercial off-the-shelf (COTS) incident name or numeric code value that expresses this incident.
Incident Class	An M-Correlator general categorization scheme used for abstractly registering interest in an incident that represents a common impact to the system. Incident types are associated with only one incident class (see Section 2.3 for details).
Description	Human-readable incident description.
Vulnerable OS and Hardware	OS type(s) and version(s), and hardware architectures required for the successful invocation of the incident.
Bound Ports and Applications	The list of required network services and applications that must be enabled on the target of an alert for this incident type to succeed.

Cluster List	One or more index values that may be associated with incident types. Two alerts that share a common cluster name may be candidates for merger should other attributes be aligned.
References	Bugtraq ID[Bug02], CERT ID [Cer02] , Common Vulnerabilities and Exposures (CVE) ID [Cve02], available descriptive URL.

Table 1 — Incident-Handling Fact-Base Field Definitions

2.2 RELEVANCE FORMULATION

M-Correlator maintains an internal topology map of the protected network, which is dynamically managed by the analyst. Automated topology map generation is supported using *Nmap* [Nma02], through which M-Correlator can identify the available assets on the network, IP address to hostname mappings, OS type and version information, active TCP and UDP network services per host, and hardware type. Nmap can be run on intervals to maintain an updated topology database, and this database can be dynamically inserted into the M-Correlator runtime process. Given both the topology database and the vulnerable OS, hardware, and bound ports fields of the incident-handling knowledge (Section 2.1), M-Correlator develops a relevance score that assesses per alert, the likelihood of successful intrusion.

As each alert is processed by M-Correlator, the associated known dependencies for that alert, as indicated within the incident handling fact base, are compared against the configuration of the target machine. Positive and negative matches against these required dependencies result in increased or decreased weighting of the relevance score, respectively. Our model for calculating asset relevance may identify as many as five attributes that match the known topology of the target host:

- OS type and version
- Hardware type
- Service suite
- Enabled network service
- Application

The relevance score is calculated on a scale from 0 to 255. 0 indicates that the incident vulnerabilities required for the successful execution of the reported security incident were *not* matched to the known topology of the target host. An unknown alert, incompletely specified dependency information in the fact base, or incomplete topology information regarding the target host, results in a neutral relevance score of 127 (i.e., the score does not contribute positively or negatively to the overall incident rank for that security incident). Scores nearer to 255 indicate that the majority of required dependencies of the reported security incident were matched to the known topology of the target host.

2.3 PRIORITY FORMULATION

The objective of mission impact analysis is to fuse related alerts into higher-level security incidents, and rank them based on the degree of threat each incident poses to the mission objectives of the target network. A *mission* is defined with respect to an administrative network domain. Mission-impact analysis seeks to isolate the highest threat security incidents together, providing the analyst with an ability to reduce the total number of incidents that must be reviewed. Abstractly, we define security incident prioritization in **Figure 3**.

$$\begin{aligned} &\text{Let} && Stream = \{e_1, e_2, e_3, \ldots, e_n\} \\ &\text{Find} && HighImpact = \{e_{\acute{a}}, e_{\hat{a}}, \ldots, e_{\emptyset}\} \subseteq Stream \\ & && \forall\ \text{Threat\_Rank}(e_i, \text{Mission}) > T_{\text{acceptable}} \\ & && e_i \in \text{HighImpact} \end{aligned}$$

Figure 3 — Security Incident Prioritization

The *mission* is the underlying objective for which the computing resources and data assets of the monitored network are brought together and used. We express this concept of mission through a *mission specification*, which is defined by the analyst. A mission specification is defined in two parts: (1) an enumeration by the analyst of those data assets and services that are most critical to the client users of the network, and (2) an identification of which classes of intrusion incidents are of greatest concern to the analyst. With respect to the critical assets and services of the protected network, the analyst must register the following items within the mission specification:

- Critical computing assets (such as file servers on which the user community depends)
- Critical network services (such as web server, a DBMS)
- Sensitive data assets (these are primarily files and directories considered highly sensitive or important to the mission of the network)
- Administrative and untrusted user accounts such as might be used by consultants

Next, the analyst can specify those intrusion incidents, or classes of incident, of greatest concern given the analyst's responsibilities within the organization. This portion of the mission specification is referred to as the *interest profile*. Interest profiles may be user specific, just as the responsibilities of analysts may be distinct. Each alert processed by M-Correlator is associated with a unique incident class type. Each incident signature listed in the incident handling knowledge base is associated with one of the following incident classes, which were derived, in part, from a review of previous work in incident classifications and vulnerability analysis [Lin98, Bak99, Ken99]:

- PRIVILEGE_VIOLATION — Theft or escalation of access rights to that of system or administrative privileges.
- USER_SUBVERSION — An attempt to gain the privileges associated with a locally administered account. This may include reports of user masquerading.
- DENIAL_OF_SERVICE — An attempt to block or otherwise prevent access to an internal asset, including host, application, network service, or system resource, such as data or a device.
- PROBE - An attempt to gain information on assets or services provided within the monitored domain.
- ACCESS_VIOLATION — An attempt to reference, communicate with, or execute data, network traffic, OS services, devices, or executable content, in a manner deemed inconsistent with the sensor's surveillance policy.
- INTEGRITY_VIOLATION — An attempt to alter or destroy data or executable content that is inconsistent with the sensor's surveillance policy.
- SYSTEM_ENV_CORRUPTION — An unauthorized attempt to alter the operational configuration of the target system or other system asset (e.g., network service configuration).

- USER_ENV_CORRUPTION — An unauthorized attempt to alter the environment configuration of a user account managed within the monitored domain.
- ASSET_DISTRESS — Operational activity indicating a current or impending failure or significant degradation of a system asset (e.g., host crash, lost service, destroyed system process, file system, or processtable exhaustion).
- SUSPICIOUS_USAGE — Activity representing significantly unusual or suspicious activity worthy of alert, but not directly attributable to another alert class.
- CONNECTION_VIOLATION — A connection attempt to a network asset that occurred in violation of the network security policy.
- BINARY_SUBVERSION — Activity representing the presence of a Trojan horse or virus.
- ACTION_LOGGED — A security relevant event logged for potential use in later forensic analyses.
- EXFILTRATION — An attempt to export data or command interfaces through an unexpected or unauthorized communication channel.

M-Correlator allows analysts to specify low, medium-low, medium, medium-high, and high interest in a particular incident type.

2.4 INCIDENT RANK CALCULATION

Incident ranking represents the final assessment of each security incident with respect to (a) the incident's impact on the mission profile as reflected by the priority calculation, and (b) the probability that the security incident reported by the INFOSEC device(s) has succeeded. Most sensors provide little if any indication regarding the outcome of an observed security incident, providing strong motivation for the production of a relevance score, where possible. It should be noted that the concept of outcome is decoupled here from that of the relevance analysis, in that outcome represents a sensor provided conclusion produced from a method unknown to the correlation engine. Relevance represents an assessment of the target system's susceptibility to an attack given vulnerability dependencies and the attack target's configuration. While both outcome and relevance may reinforce each other in increasing an overall incident rank score, so too can they neutralize each other in the face of disagreement.

Once a mission profile is specified, security incidents may be assessed and ranked against the profile. We concisely define incident ranking as illustrated in **Figure 4**.

Incident Rank: An assessment and ranking of events {e1,. ..., en, …}

with respect to { *Mission Profile* = $\{CR_{assets}, CR_{resources}, CR_{users}, Incident_{weight}\}$ &

Probability of Success → { Alert Outcome, Relevance}

Figure 4 — Incident Rank Calculation

2.4.1 The Bayes Calculation

Mathematically, relevance, priority, and incident rank calculations are formulated using an adaptation of the Bayes framework for belief propagation in trees, described in [Pea88] and [Val00]. In this framework, belief in hypotheses at the root node is related to propagated belief at other nodes and directly observed evidence at leaf nodes by means of conditional probability tables (CPTs). At each node, "prior"

probabilities $p(parent)$ are propagated from the parent, and "likelihoods" $l(child)$ are propagated to the parent. The branch and node structure of the tree expresses the three major aspects of the calculation, namely, outcome, relevance, and priority.

Bayes networks compute belief in a number of hypothesis states. In our adaptation, the root node considers the hypothesis "criticality" and states "low", "medium", and "high". A mapping function transforms this to a single value on a scale of 0 to 255.

The predefined CPTs encode the mathematical relationship between observable evidence and derived intermediate node values to the overall criticality of the alert with respect to the mission. Our predefined CPTs have been developed through extensive experience and experimentation. However, we recognize the need to adapt the framework for specific environments. To this end, we include an adaptive mode wherein the analyst presents simulated alerts, which are ranked by the system. At this time the analyst either accepts the outcome or enters a desired ranking. This causes the CPTs to adapt slightly in order to more accurately reflect the administrator's preference. The adaptation occurs with no knowledge of the underlying Bayes formalism on the part of the administrator. The analyst may optionally revert to the original CPT values as well.

2.4.2 The Rank Tree

Figure 6 represents the complete incident rank tree, which brings together the contributions of alert outcome (when provided by the INFOSEC device), relevance score, and security incident priority score. These three contributors are represented by the three major branches of the incident rank tree. The priority subtree represents a merger of the incident class importance, as defined by the analyst, and the criticality of the attack target with respect to the mission of the network. The elements of the respective CPTs reflect $P(criticality = c \mid priority = p)$. Each of these matrices represents two values of criticality by

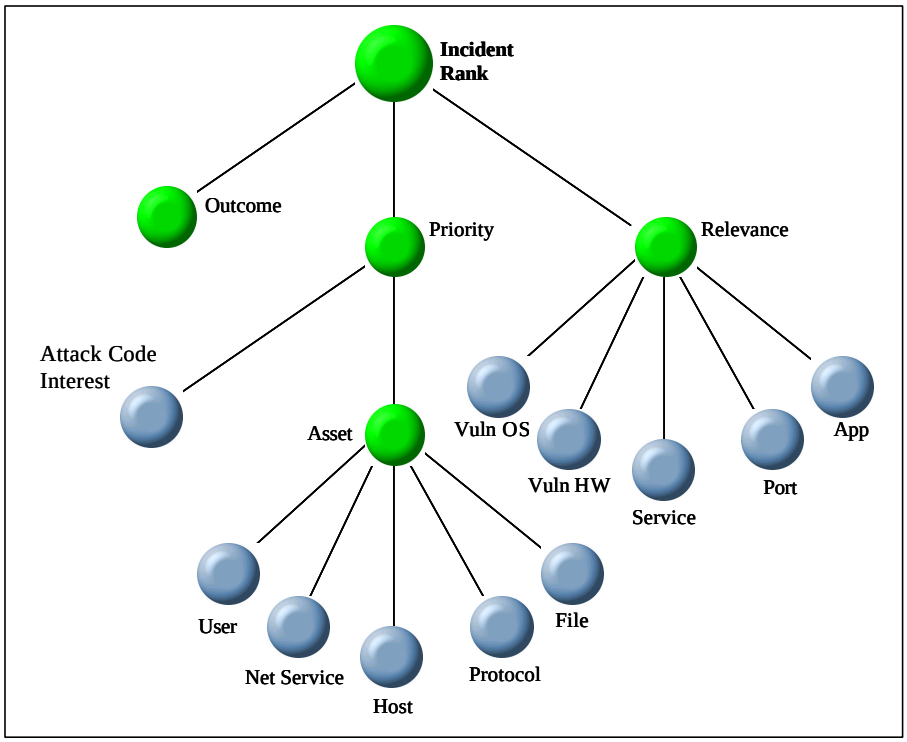

Figure 5 — Incident Rank Calculation

three values of priority. Therefore, the local knowledge base consists of a set of CPTs linking the attribute to the appropriate node on its main branch. If the attribute is not observed in a given alert, the state of the corresponding node is not changed, and thus this attribute does not influence the result one way or the other. If this attribute is observed in a subsequent update for the same alert, our system adjusts the previous prioritization for the new information.

As discussed in Section 2.2, our model identifies five equally weighted contributing attributes that formulate the relevance score: vulnerable OS, vulnerable hardware, service suite, bound ports, and application. The relevance subtree in **Figure 5** illustrates these elements. Again, the Bayes net is robust in cases where the alert does not provide values for all these attributes.

3. ALERT CLUSTERING ALGORITHM

M-Correlator employs an alert clustering algorithm, which is used to consolidate both network and host-based INFOSEC alerts that occur in close (dynamically adjustable) temporal proximity into correlated security-incident reports. INFOSEC alerts regarding network communications are merged through an analysis of common network session, as defined by port and IP address matches, and common observer, alert type, or, more liberally, by common alert classification as defined in the incident handling fact base. INFOSEC alerts regarding host activity are merged through an analysis of common session, as defined through user session attributes such as process ID or user ID, common observer, alert type, or more liberally by common alert classification.

Figure 7, shows an example M-Correlator clustering policy. (Note that we are deliberately restricting this discussion to the more straightforward properties of these policies.) Given a new security-incident report, the M-Correlator first determines if the report is a candidate for the policy. In this example, if the report originates from either a network sensor or a host-based sensor in which the source process ID and source user names are known, then the report is a candidate for further processing.

The clustering policy's Match_If clause defines the criteria by which reports are clustered. Thus, in this case, all clustered report incident signatures and their observer stream identifiers (if extant) must match. Also, if the sensor is network-based, the reports must have matching source and target IP addresses; while if host-based, the reports must have matching target IP addresses, and, if extant, matching source process IDs and source user names.

A clustering policy also specifies the longevity of a clustered report when there are no subsequent candidate reports; the delay before a clustered report is initially issued, and the refresh interval between clustered report updates, again whose purpose is to reduce report traffic.

The incident-handling fact base also supports the specification of a set of attributes that represent loose relationships among alerts of different classes. For example, consider a firewall that reports a series of connection violations between external host A and internal host B, and suppose this activity temporally overlaps an internal network IDS report of a port sweep on several ports. That is, the port sweep manifested itself by two sets of activity: (1) connection attempts that were blocked by the firewall filtering policy, and (2) connections that were allowed through the firewall, but in aggregate caused the network IDS to consider the flurry of connections as a potential port scan. Alert clustering tags are established by the incident handling fact base maintainer, and allow M-Correlator a greater ability to leverage associations unique to specific known scenarios. In this example, a shared cluster name within the incident-handling fact base allows M-Correlator to merge the connection violation reports with the port scan alerts.

```
Profile                 Cross_Sensor_Signature_And_Session_Match
Policy                  Liberal

Candidate_If            [ OR
        [ IN_GROUP              observer_name Network_Sensors ]
        [ AND
            [ IN_GROUP          observer_name Host_Sensors ]
            [ NOT
                [ AND
                    [ NULL source_pid ]
                    [ NULL source_username ]
                ]
            ]
        ]
    ]

Match_If [ AND
                [ EQ            incident_signature ]
                [ NULL_OR_EQ    observer_stream ]
                [ OR
                        [ AND
                             [ IN_GROUP observer_name Network_Sensors ]
                             [ ELEMENTS_EQ            source_IParray ]
                             [ ELEMENTS_EQ            target_IParray ]
                        ]
                        [ AND
                             [ IN_GROUP observer_name Host_Sensors ]
                             [ ELEMENTS_EQ            target_IParray ]
                             [ NULL_OR_EQ             source_pid ]
                             [ NULL_OR_EQ             source_username ]
                        ]
                ]
         ]

Delay_Until_Expire      600
Delay_Until_Flush       90
Initial_Flush_Delay     90

Enable                  true
Unique_Match            true
Merge_Action            fuse
```

Figure 7 — Example Alert Cluster Policy Specification

4. AN EXAMPLE MISSION SPECIFICATION

A brief example mission specification is subsequently used here to illustrate mission-based impact analysis. This example is based on a simulated heterogeneous network, illustrated in **Figure 6**. The network consists of hosts employing four different operating systems and protected by several distributed INFOSEC devices. Four Sun Solaris systems are protected by host-based intrusion detection sensors (SRI's EMERALD eXpert-BSM [Lin01]), and three network intrusion detection systems (eBayes-TCP [Val00], eXpert-Net, and RealSecure). Traffic entering the LAN is filtered through a Checkpoint firewall, whose alert reporting mechanism is wrapped to forward INFOSEC alerts to the M-Correlator.

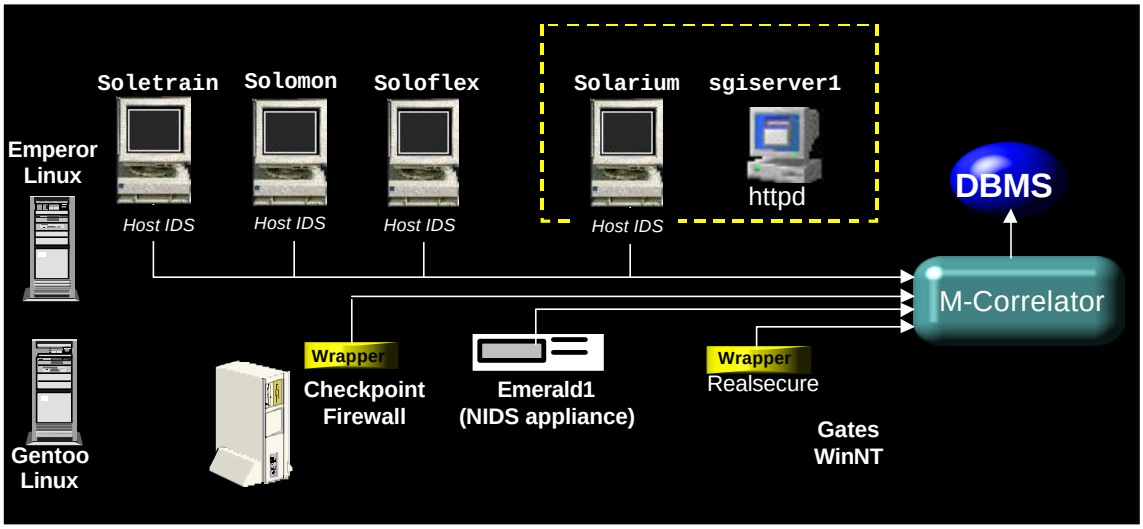

Figure 6 — An Experimental Simulation Network

Within this experimental LAN, there is one mission-critical server, Solarium, which operates as a file-server for the rest of the network, and one mission-critical SGI HTTP server. There are two administrative accounts for this LAN, em_admin1 and em_admin2, and three untrusted consultant accounts that the administrators want to monitor closely. A highly sensitive and valuable source code directory is located on host Solomon. With respect to administrative responsibilities, the analyst in this case study is most concerned with rapidly responding to potential subversion of system control, through either privilege theft or modification of the systems themselves. The analyst is also concerned with attacks against users, but to a lesser extent than system-directed attacks that may have direct impact on the security of all user accounts. The analyst, who is responsible for maintaining availability and watching for suspicious probes and behaviors, feels that at this time alerts of these types should receive a slightly lower priority than direct attacks on the system and its users. **Table 2** provides an example mission specification based on the experimental LAN in **Figure 6**.

```
Critical_Assets [
    :solarium:
    TCP:sgiserver1:http
 ]
```

```
Critical_Resources [
  em_admin1:192.12.34.0/24
  em_admin1:192.12.34.0/24
  consultant1:192.12.34.0/24
  consultant2:192.12.34.0/24
  consultant3:192.12.34.0/24
  :/proprietary/src/:solomon
 ]
```

```
Interest_Policy [

 PRIVILEGE_VIOLATION    High
 SYSTEM_ENV_CORRUPTION  High
 ACCESS_VIOLATION       High
 BINARY_SUBVERSION      High
 CONNECTION_VIOLATION   Medium
 USER_ENV_CORRUPTION    Medium
 USER_SUBVERSION        Medium
 INTEGRITY_VIOLATION    Medium
 EXFILTRATION           Medium
 PROBE                  Low
 SUSPICIOUS_USAGE       Low
```

Alert_Filters [*__empty__*]	ACTION_LOGGED Low DENIAL_OF_SERVICE Low ASSET_DISTRESS Low MinAnomalyScore 90 MinConfidenceScore 30]

Table 2 — Sample Mission Specification

The top left quadrant defines the list of host and network services that are critical to the mission of the example network. The middle left quadrant enumerates both critical user accounts and critical host files or directories. The syntax for critical asset and resource specifications is as follows:

Host_specifier ::= (<IPaddress> | <host_name>) ['/' <num_mask_bits>]
Port_list ::= '[' <port> {<port>} ']'
Service_specifier ::= [<proto> ':'] [Host_specifier] [':' (<port> | Port_list)]
Resource_specifier ::= [(<user_name> | <uid>) ':']
[<path>] [':' Host_specifier]

The right quadrant defines the analyst's level of interest in the various classes of security alerts. In this scenario, the analyst is primarily concerned with direct threats of privilege and access violations, while being much less concerned with issues of probes, availability, and other nonspecific suspicious activity. In addition, the interest profile allows the analyst to define a minimum anomaly score from anomaly reporting services and minimum confidence scores provided by probability-based intrusion detection services, such as those described in [Val00].

4.1 A BASIC RANK SCENARIO

Table 3 is an incident-ranking scenario that demonstrates the concepts of Section 3, based on the sample environment presented in Section 4. In this example, four INFOSEC devices (EMERALD eXpert-Net, eXpert-BSM, Checkpoint Firewall-1, and ISS RealSecure) contribute to a stream of eleven independent security incidents. Based on the sample mission profile defined in Section 4, the incident column entries that are recognized as high interest are listed in bold, and column entries of medium interest are underlined. Critical servers and untrusted source users, as defined in the sample mission profile, are also listed in bold.

The Active Port and OS/Architecture columns identify contributions of the relevance scores. "Yes" in the Active Port and OS/Architecture columns indicates that the dependent network services, operating system, and hardware required for the successful execution of the given alert were indeed found to be present and enabled on the target machine.

Observer	Incident	Source IP	Target IP	Src user	Active Port	Outcome	OS/Arch
eBSM-Solarium	ROOT_CORE_CREAT	200.55.19.143.100	solarium.23	**consultant1**	N/A	**success**	**Yes**
eBSM-Soloflex	ROOT_CORE_CREAT	200.55.19.144.3467	gates.23	**consultant1**	N/A	fail	N/A
eXpert-Net	FTP_CWD_PROBE	200.55.19.145.4125	gentoo.21	**consultant1**	No	**success**	N/A
eXpert-Net	FTP_CWD_PROBE	200.55.19.146.3341	gentoo.21	anonymous	No	fail	N/A
eXpert-Net	FTP_CWD_PROBE	200.55.19.147.5143	emperor.21	anonymous	**Yes**	**success**	N/A
Checkpoint	TCP_CONN_DENIED	200.55.19.148.1657	gates.53		No	fail	No
ISS-realsecure	NFSMKNOD	200.55.19.149.1235	sgiserver.53		No	**success**	No
eBSM-Solarium	PRIVATE_FILE_ALT	200.55.19.150.1809	**solarium.53**	**consultant1**	No	fail	No
eBSM-Solarium	PIRVATE_FILE_ALT	200.55.19.151.5413	**solarium.53**	**consultant1**	No	**success**	No
ISS-realsecure	**HTTP_SGI_WRAP**	200.55.19.152.1243	**sgiserver1.80**	nobody	**Yes**	fail	**Yes**
ISS-realsecure	**HTTP_SGI_WRAP**	200.55.19.151.3467	**sgiserver.80**	**consultant1**	**Yes**	**success**	**Yes**

Table 3 — Incident Rank Example Set

Table 4 illustrates the view of the **Table 3** dataset from the perspective of the incident rank calculation process, and the resulting ranking produced by M-Correlator (1 represents the highest threat ranking and 11 represents the lowest threat ranking). Alerts of highest threat share the property that their combined contributing elements are of high priority, relevance, and positive outcome.

4.2 A MULTI-SENSOR ALERT SCENARIO

As a further example of incident ranking in combination with M-Correlator's alert clustering algorithm, the following is a multi-INFOSEC device alert scenario. Like the basic scenario, this scenario is based on the experimental environment discussed in Section 4. **Table 5** provides a sampling of alerts produced from five INFOSEC devices distributed in the M-Correlator experimental LAN: EMERALD eBayes-TCP and eXpert-Net network intrusion detection sensors, Checkpoint Firewall-1, ISS Realsecure, and eXpert-BSM.

		priority		relevance		outcome
Ranking	**Incident**	**Target IP**	**Src user**	**Active Port**	**OS/Arch**	**Outcome**
4	ROOT_CORE_CREAT	**solarium.23**	**consultant1**	N/A	**Yes**	**success**
10	ROOT_CORE_CREAT	gates.23	**consultant1**	N/A	N/A	fail
7	FTP_CWD_PROBE	gentoo.21	**consultant1**	No	N/A	**success**
11	FTP_CWD_PROBE	gentoo.21	anonymous	No	N/A	fail
6	FTP_CWD_PROBE	emperor.21	anonymous	**Yes**	N/A	**success**
9	TCP_CONN_DENIED	gates.53		No	No	fail
5	NFS_MKNOD	sgiserver.53		No	No	**success**
8	PRIVATE_FILE_ALT	**solarium.53**	**consultant1**	No	No	fail
3	PRIVATE_FILE_ALT	**solarium.53**	**consultant1**	No	No	**success**
2	**HTTP_SGI_WRAP**	**sgiserver1.80**	nobody	**Yes**	**Yes**	fail
1	**HTTP_SGI_WRAP**	**sgiserver.80**	**consultant1**	**Yes**	**Yes**	**success**

Table 4 — Incident Rank Resolution

In this scenario, 79 alerts are forwarded to M-Correlator for analysis and database storage. **Table 5** shows the time at which each INFOSEC alert was generated, the INFOSEC device that generated the alert, the alert type, the source and destination of the attack, and alert outcome and relevance calculation. The alert is identified as relevant if M-Correlator was able to confirm that at least one of the alert dependencies in the relevance calculation was found to match. The alert is identified as nonrelevant if no alert dependencies are found to match, and N/A indicates that the relevance calculation could not be performed because of lack of information from the incident-handling fact base or regarding the target host. Bold text is used in **Table 5** to indicate that the value represents a critical item in the sample mission specification of Section 4.

Table 6 presents the alert stream represented in **Table 5** ranked and aggregated by the M-Correlator prototype. In this example, **Table 5**'s 79 original INFOSEC alerts were processed by M-Correlator and stored in an Oracle database. The security incidents shown in **Table 6** were merged and ranked as follows:

9:41am	eBayes-TCP	Port_Scan	200.55.19.100 --> gates.[21 22 23]	Success, N/A
9:41am	Checkpoint	TCP_Connect_Violation	200.44.19.100 --> gates	Failed, N/A
		x 70 TCP_Connect_Violations		
9:45am	Realsecure	Kerberos_User_Snarf	195.16.19.56 --> emperor	Unknown, Non-Relevant
9:48am	eBayes-TCP	Port_Scan	200.55.19.100 --> gates.[21 22 23 79 80]	Success, N/A
9:51am	Realsecure	**Intel_Buffer_Overflow**	200.55.19.149.3450 --> gentoo.143	Unknown, **Relevant**
9:51am	eXpert-Net	**Imap_Overflow**	200.55.19.149.3450 --> gentoo.143	Success, **Relevant**
9:52am	eBayes-TCP	Port_Scan	200.55.19.100 --> Gates.[21 22 23 79 80 514]	Success, N/A
10:02am	eXpert-BSM	**Buffer_Overflow**	console --> **solarium**	Success, **Relevant**
10:04am	eXpert-BSM	Illegal_File_Alteration	console --> **solarium**	Success, **Relevant**
10:05am	eXpert-BSM	Illegal_File_Alteration	console --> **solarium**	Success, **Relevant**

Table 5 — Cross Sensor Incident Rank and Aggregation Dataset

1. Entry 1: This incident represents the three eXpert-BSM reports of malicious activity from the same user session. The aggregate priority of these alerts was high because this incident included a high-interest privilege subversion attack on the critical fileserver, Solarium. The attacks were also successful, and all were found relevant to the target machine.
2. Entry 2: The second-highest-ranked incident represents ISS Realsecure's Intel-Buffer-Overflow and eXpert-Net's Imapd-Overflow alerts. These alerts both shared the common alert class privilege-subversion and common network session attributes. While this security incident was registered in the mission specification as high interest, and found both successful and relevant, the alert was performed against a lower-interest target host.
3. Entry 3: The M-Correlator used its alert clustering algorithm to merge the eBayes-TCP Port-Scan alert with the Checkpoint Firewall Connection-Violation alerts. These alerts pertain to the same source and target addresses, and share a common alert cluster tag called Generic-Probe. The alerts are ranked low because of the analyst's low interest in probing activity, and because the alert did not target a critical asset or resource.
4. Entry 4: The lowest-ranked security incident is a Realsecure Kerberos-User-Snarf probe. This alert represents a distinct security incident, was evaluated overall as low interest because the at-

tack targeted a noncritical asset, and apparently represented a false alarm, as there is no Kerberos server present on the target machine.

Rank	Time	Incident	Connection	Observers	Other
1	10:02am - 10:05am	**Buffer_Overflow:** **- buffer_overflow** **- illegal_file_alteration** **- illegal_file_alteration**	Console --> **Solarium**	eXpert-BSM	**Success, Relevant**
2	9:51am	**Generic_Priv_Subv:** **- imapd_overflow** **- intel_buff_overflow**	200.55.19.149 --> gento.143	RealSecure, eXpert-Net	**Success, Relevant**
3	9:41am - 9:52am	Generic_Probe: - Port_Scan - TCP_Connect_Violation	200.55.19.100 --> gates.[21 22 23 79 80 514]	eBayes-TCP Checkpoint	**Success**
4	9:45am	**Kerberos_User_Snarf**	195.16.19.56 --> emperor	RealSecure	Unknown Irrelevant

Table 6 — Ranked and Aggregated Security Incidents

5. RELATED RESEARCH

The broad problem of information security alert management and post-sensor analysis is an area that has been undergoing a great deal of activity. In the commercial space, one issue receiving particular attention is that of sensor overload through alert fusion and better methods for visualization. Some systems offer a form of severity measurement that attempts to prioritize alerts. However, unlike M-Correlator, which attempts to formulate a security incident ranking based on a multi-attribute mission profile specification, the vast majority of severity metric services rely on hard-coded mappings of attributes to fixed severity indexes.

There is also a growing momentum of research activity to explore intrusion report correlation as a separable layer of analysis above intrusion detection. In the space of alert aggregation and false positive reduction there are a number of ongoing projects. These include work by Honeywell, which is developing Argus, a qualitative Bayesian estimation technology to combine results from multiple intrusion detection systems [Gei01]. The work to date has emphasized false-positive reduction through *a priori* assessments of sensor reporting behavior. M-Correlator also attempts to reduce false positives, but through an emphasis on the relevance of an alert to the target system's configuration.

IBM Zurich [Deb01] is also exploring common attribute-based alert aggregation in the Tivoli Enterprise Console, as are Columbia University and Georgia-Tech using association rules [Lee00]. Onera Toulouse is using an expert-system-based approach for similarity formulation [Cup01], and SRI International is using a probabilistic-based approach to attribute similarity recognition [Val01]. M-Correlator's alert clustering algorithm is very similar in purpose and outcome to these systems as it, too, attempts to perform alert fusion in the presence of incident type disagreement, feature omission, and competing incident clustering opportunities.

Another major thrust of activity involves the development of complex (or multistage) attack modeling systems, capable of recognizing multistage (or multi-event) attack scenarios within the stream of distrib-

uted intrusion detection sensors. Stanford has developed extensions to its ongoing research in Complex Event Processing (CEP) in the domain of intrusion report correlation [Per00]. In this work, complex multistage scenarios can be represented in concept abstraction hierarchies using the CEP language. This approach provides methods for efficient event summarization and complex pattern recognition. The Stanford work differs significantly from M-Correlation in that it emphasizes scenario recognition and content summarization, whereas M-Correlation emphasizes impact analysis, where the impact is defined with respect to the mission objectives of the target network. In addition, IET Incorporated is involved in research to develop situation-aware visualization software for managing security information and recognizing composite attack models [Amb01]. There is an emphasis on large-scale attack recognition using a probabilistic domain-modeling algorithm. The effort differs significantly from M-Correlation in that it too emphasizes model recognition. Finally, UC Santa Barbara's STAT-based web of sensors work is exploring the use of the STATL language as a basis for complex attack modeling [Gio01].

6. CONCLUSION

We have discussed a mission-impact-based approach to alert prioritization and aggregation. This research has led to the development of a prototype system called M-Correlator, which is capable of receiving security alert reports from a variety of INFOSEC devices. Once translated to an internal incident report format, INFOSEC alerts are augmented, and, where possible, fused together through a chain of processing. A *relevance score* is produced through a comparison of the alert target's known topology against the vulnerability requirements of the incident type, which is provided to M-Correlator by an *Incident Handling Fact Base*. Next, a *priority calculation* is performed per alert to indicate (a) the degree to which the alert is targeted at critical assets, and (b) the amount of interest the user has registered for this alert type. Last, an overall *incident rank* is assigned to each alert, which brings together the priority of the alert with the likelihood of success.

Once ranked, the M-Correlator attempts to combine related incident alarms with an attribute-based *alert clustering algorithm*. The resulting correlated incident stream represents a filtered, lower-volume, content-rich security-incident stream, with an incident-ranking scheme that allows the analyst to identify those incidents that pose the greatest risk to the monitored network.

M-Correlator has reached a maturity level where trial releases of the system will begin this year in several computing environments with the U.S. Department of Defense. In time, we believe that mission-impact based analyses will prove themselves extremely useful both to human analysts, and to other consumers, such as automated response technology that must sift through thousands of alerts daily in search of alarms worthy of proactive response. Extension of the basic M-Correlator algorithm is already underway to incorporate its logic into a real-time response engine.

REFERENCES

[Amb01] D'Ambrosio, B, M. Takikawa, D. Upper, J. Fitzgerald, and S. Mahoney, "Security Situation Assessment and Response Evaluation," *Proceedings (DISCEX II) DARPA Information Survivability Conference and Exposition*, Anaheim, CA, Vol. I, June 2001.

[Bak99] D.W. Baker, S.M. Christey, W.H. Hill, and D.E. Mann, "The Development of a Common Enumeration of Vulnerabilities and Exposures," *Proceedings of the Second International Workshop on Recent Advances in Intrusion Detection (RAID)*, September 1999.

[Bug02] Bugtraq. Security Focus Online. http://online.securityfocus.com/archive/1

[Cer02] CERT Coordination Center. Cert/CC Advisories Carnegie Mellon, Software Engineering Institute. Online. http://www.cert.org/advisories/

[Cup01] F. Cuppens, "Managing Alerts in a Multi-Intrusion Detection Environment," *Proceedings 17th Computer Security Applications Conference,* New Orleans, LA, December 2001.

[Cve02] Common Vulnerabilities and Exposures. The MITRE Corporation. http://cve.mitre.org/

[Deb01] H. Debar and A. Wespi, "Aggregation and Correlation of Intrusion-Detection Alerts," *Proceedings 2001 International Workshop on Recent Advances in Intrusion Detection (RAID),* Davis, CA, October 2001.

[Gio01] G. Vigna, R.A. Kemmerer, and P. Blix, "Designing a Web of Highly-Configurable Intrusion Detection Sensors," *Proceedings 2001 International Workshop on Recent Advances in Intrusion Detection (RAID),* Davis, CA, October 2001.

[Gei01] C.W. Geib and R.P Goldman, "Probabilistic Plan Recognition for Hostile Agents," *Proceedings of FLAIRS 2001 Special Session on Uncertainty* - May 2001.

[Kah99] C. Kahn, P.A. Porras, S. Staniford-Chen, and B. Tung, "A Common Intrusion Detection Framework," http://www.gidos.org.

[Ken99] K. Kendall, "A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems," Master's Thesis, Massachusetts Institute of Technology, June 1999.

[Lee00] W. Lee, R.A. Nimbalkar, K.K. Yee, S.B. Patil, P.H. Desai, T.T. Tran, and S.J. Stolfo, "A Data Mining and CIDF-Based Approach for Detecting Novel and Distributed Intrusions", *Proceedings 2000 International Workshop on Recent Advances in Intrusion Detection (RAID),* Toulouse, France, October 2000.

[Lev01] D. Levin, Y. Tenney, and H. Henri, "Issues in Human Interaction for Cyber Command and Control," *Proceedings (DISCEX II) DARPA Information Survivability Conference and Exposition,* Anaheim, CA, Vol. I, June 2001.

[Lin01] U. Lindqvist and P.A. Porras, "eXpert-BSM: A Host-based Intrusion Detection Solution for Sun Solaris," *Proceedings 17th Computer Security Applications Conference,* New Orleans, LA, December 2001.

[Lin98] U. Lindqvist, D. Moran, P.A. Porras, and M. Tyson, "Designing IDLE: The Intrusion Detection Library Enterprise," *Proceedings 1998 International Workshop on Recent Advances in Intrusion Detection (RAID),* Louvain-la-Neuve, Belgium, September 1998.

[Nma02] NMAP Network Mapping tool. http://www.insecure.org/nmap/

[Pea88] Pearl, J. "Probabilistic Reasoning in Intelligent Systems," Morgan-Kaufmann (1988).

[Per00] L. Perrochon, E. Jang, and D.C. Luckham.: Enlisting Event Patterns for Cyber Battlefield Awareness. *DARPA Information Survivability Conference & Exposition (*DISCEX'00), Hilton Head, South Carolina, January 2000.

[Por97] P.A. Porras and P.G. Neumann, "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," *Proceedings National Information Systems Security Conference*, NSA/NIST, Baltimore, MD, October 1997. http://www.sdl.sri.com/emerald/emerald-niss97.html

[Roe99] M. Roesch, "Lightweight Intrusion Detection for Networks," *Proceedings of the 13th Systems Adminstration Conference — LISA 1999,* November, 1999. www.snort.org/docs/lisapaper.txt

[Val00] A. Valdes and K. Skinner, "Adaptive, Model-based Monitoring for Cyber Attack Detection", *Proceedings 2000 International Workshop on Recent Advances in Intrusion Detection (RAID)*, Toulouse, France, October 2000.

[Val01] A. Valdes and K. Skinner, "Probabilistic Alert Correlation," *Proceedings 2001 International Workshop on Recent Advances in Intrusion Detection (RAID)*, Davis, CA, October 2001.