Mitigating Attacks Against Measurement-Based Adaptation Mechanisms in Unstructured Multicast Overlay Networks

AAron Walters David Zage Cristina Nita-Rotaru
Department of Computer Science and CERIAS, Purdue University
305 N. University St., West Lafayette, IN 47907 USA
{arwalter,zagedj,crisn}@cs.purdue.edu

***Abstract*— Many multicast overlay networks maintain application-specific performance goals such as bandwidth, latency, jitter and loss rate by dynamically changing the overlay structure using measurement-based adaptation mechanisms. This results in an unstructured overlay where no neighbor selection constraints are imposed. Although such networks provide resilience to benign failures, they are susceptible to attacks conducted by adversaries that compromise overlay nodes. Previous defense solutions proposed to address attacks against overlay networks rely on strong organizational constraints and are not effective for unstructured overlays. In this work, we identify, demonstrate and mitigate insider attacks against measurement-based adaptation mechanisms in unstructured multicast overlay networks. The attacks target the overlay network construction, maintenance, and availability and allow malicious nodes to control significant traffic in the network, facilitating selective forwarding, traffic analysis, and overlay partitioning. We propose techniques to decrease the number of incorrect or unnecessary adaptations by using outlier detection. We demonstrate the attacks and mitigation techniques in the context of a mature, operationally deployed overlay multicast system, ESM, through real-life deployments and emulations conducted on the PlanetLab and DETER testbeds, respectively.**

Keywords: Overlay Networks, Security, Insider Attacks, Adaptivity

I. Introduction

Multicast overlay networks were proposed as a viable application level multicast architecture to overcome the scarcity of native IP multicast deployments. Examples of such networks include ESM [1], Nice [2], ALMI [3], and Overcast [4]. Moving buffering and relaying functionality from core routers to end-systems provides support for easy deployment and increased scalability. In addition, using dissemination structures constructed based on partial overlay topology information allows for reduced overhead.

Many multicast overlay networks optimize application-specific performance goals such as bandwidth, latency, jitter, and loss rate by dynamically adapting the overlay topology. This improves suboptimal overlay meshes resulting from random initial neighbor selection, aggressive partition repair, group membership changes, and transient conditions in the underlying physical network. Each node maintains partial overlay topology in the form of a set of neighbor nodes and an upstream node. A node changes its upstream node if the performance becomes inadequate by monitoring its performance from the multicast source and periodically probing its neighbor nodes about their performance. We refer to this process as *adaptation* and to the mechanisms used to achieve it as *adaptation mechanisms*. There are no constraints in the selection of the neighbor set and no imposed constraints in the resulting overlay. Such networks are referred to as *unstructured overlay networks* to differentiate them from *structured overlay networks* [5], where the overlay topology offers predefined bounds and organizational invariants by constraining the set of nodes eligible to become neighbors of a given node. Examples of multicast systems using structured overlay networks include Scribe [6] and SplitStream [7].

While pushing functionality to end-systems allows overlay networks to achieve better scalability, it also makes them vulnerable as trust is pushed to the fringes of the Internet where end-nodes are more likely to be compromised than core routers [8]. Overlay networks are more susceptible to insider attacks conducted by attackers that infiltrate the overlay or compromise some of its nodes. One attack that does not require significant work from the attacker is to exploit the adaptation mechanisms by influencing the accurate interpretation of performance observations, and the correctness of the responses received from probed nodes. As a result, an attacker can influence the overlay construction and maintenance, controlling a significant part of the traffic. This facilitates further attacks such as selective data forwarding, cheating, traffic analysis, and overlay partitioning. Some attacks, such as selective forwarding, may ultimately be noticed by the victim so they can be effectively addressed by deploying a posteriori detection mechanism. Other attacks, such as traffic analysis, do not have immediately observable results. It is thus critical to address the primary attacks that allow the adversary to control the overlay structure maintenance.

Previous work addressing malicious attacks on overlay networks focused on structured overlays [9], [10], [11], [12], [13], [14] used for file sharing applications. In this case, the attacker controls the file discovery by manipulating the control and data messages routed within the overlay, poisoning the routing table, or partitioning the network. The proposed mitigation tech-

1-4244-0593-9/06/$20.00 ©2006 IEEE

niques leverage the strong organizational constraints imposed on neighbor selection and the invariant relationships between neighbors. While solutions for attacks in structured overlay networks offer valuable insights into the problem space, they are not appropriate for unstructured overlay multicast networks where no structural constraints exist between neighbors.

In this paper, we focus on identifying, demonstrating, and mitigating insider attacks in unstructured multicast overlay networks. The attacks exploit adaptation mechanisms that these networks use in order to maintain application-specific performance. Current adaptation mechanisms assume that the information reported by probed nodes is always correct and fail to take into account the effects of malicious attackers on their surrounding environment. Unlike previous work demonstrating attacks exploiting adaptivity [15], [16], our work considers the effects of insider adversaries in the context of overlay networks. We summarize our key contributions:

• We provide a characterization of the types of mechanisms currently used to achieve adaptivity in overlay networks and identify attacks against these mechanisms. We refer to the attacks as *attraction*, *repulsion*, and *disruption*.

• We provide an analysis of the solution space for mitigating insider attacks that exploit measurement-based adaptation: preventing incorrect or unnecessary adaptations, increasing stability by incorporating metrics that reflect stability into the decision process, detecting observable malicious behavior such as degradation of service, and isolating the malicious nodes.

• We propose techniques to reduce incorrect and unnecessary adaptations by using spatial and temporal correlations to perform context-sensitive outlier analysis. A key component of our solution is based on the observation that several estimated metrics are dependent variables and the overlay and multicast logical networks share overlapping physical links.

• We demonstrate the effectiveness of the identified attacks and the benefits of our defense mechanisms in the context of a well-known and operationally deployed multicast system, ESM [1], through experiments and emulations conducted on the PlanetLab [17] and DETER [18] testbeds, respectively.

Roadmap: The rest of the paper is organized as follows. We specify our system and attack models in Section II. We discuss adaptation mechanisms employed by overlay networks and identify attacks against them in Section III. We propose defense mechanisms in Section IV. We present experimental results demonstrating the attacks and the defense techniques in Section V. We overview related work in Section VI and conclude our work in Section VII.

II. System and Attacker Model

A. System Model

We focus on overlay networks providing support for single-source broadcasting applications that are high-bandwidth (hundreds of kilobits per second) and real-time, but not interactive. The system consists of a set of nodes and a data source node communicating via unicast links. All nodes but the source have similar functionality. The nodes are not only receivers of data, but also contribute to the routing process. The source is assumed to be continually available.

The overlay construction is self-organized and distributed. Each node maintains a neighbor set, a routing table and the upstream node forwarding the data, referred to as the node's *parent*. The neighbor set represents only partial topology information and consists of nodes that are currently reachable in the overlay. The nodes in the neighbor set are referred to as *peers*. No node has complete knowledge of the dissemination topology. The neighbor set is bootstrapped at join time by contacting the source and is continually updated via a membership protocol. There are no constraints placed on the members of a node's neighbor set. The routing table represents a set of nodes that the node is responsible for routing data to, also referred to as *children*. The size of this set is limited by a system characteristic called *saturation degree*, representing the number of concurrent data streams the node is able to support before saturating the underlying physical network link.

Each node maintains a set of performance variables for each member of its neighbor set. These variables are dictated by application-specific goals and are continuously measured by using passive observation and active probes. A node uses the collected performance metrics to select a new parent from its neighbor set if the performance becomes inadequate.

B. Attacker Model

We consider a constrained-collusion Byzantine adversary model similar to that proposed in [12], with a system size of N and a bounded percentage of malicious nodes f ($0 \leq f < 1$) behaving arbitrarily. The set of malicious nodes is partitioned into disjoint coalitions with intra-coalition cooperation possible. We assume a malicious adversary has access to all data at a node as any legitimate user would (insider access), including cryptographic keys stored at a node. This access can be the result of the adversary bypassing the authentication mechanisms or compromising a node through other means. Nodes cannot be completely trusted although they are authenticated. We assume that data authentication and integrity mechanisms are deployed and we focus only on attacks directed at the adaptation mechanisms. We assume the source is trusted and cannot be compromised.

III. Attacks Exploiting Measurement-Based Adaptation in Overlay Networks

Any adaptive network protocol based on measurements involves periodically observing and estimating the network conditions, followed by making an adaptation decision. For multicast overlays, the variables that are observed and estimated include latency, jitter, bandwidth, and loss rate. The adaptation decision consists of a node selecting a new parent by weighing the associated costs versus benefits that could occur as the result of the adaptation quantified through a utility function [19]. For unstructured overlays, there are no structural constraints placed on this selection.

Previous work studied the quality of the data observation and estimation, as well as the ability of the metrics to

accurately reflect the state of the network. Examples of factors that influence data quality include data freshness, variability and the presence of noise. Mechanisms proposed to address these issues are data sampling [20], data smoothing [21], metric construction [22], as well as data summarization and aggregation [23]. Previous work also studied instabilities [24], [25], [26], such as the oscillatory behavior commonly referred to as flapping, occurring when nodes rapidly switch between seemingly equal alternatives. New techniques such as utility discretization [26], [27], randomization [26], [28], damping [25], and hysteresis [28], [20] were deployed to mitigate these phenomena and provide a tradeoff between responsiveness to change and instability.

None of the mechanisms described above take into account adversarial environments, since they only address the effects of benign problems. However, compromised overlay nodes can take advantage of the adaptation process to gain control over overlay traffic by manipulating the path selection or the overlay topology. We classify these attacks as *attraction attacks*, *repulsion attacks*, and *disruption attacks*. Any of these attacks can be conducted by an adversary by lying about its observed performance metrics or by artificially influencing the performance metrics observed by other nodes.

Attraction attacks are a form of "bait-and-switch" attacks, where a malicious node manipulates the observed data in order to present the network conditions as better than they are. The attack can also target one particular node, in which case the attacker persuades the victim to attach to a malicious parent in the dissemination structure. The final goal of the attack can be manipulating data, performing traffic analysis, performing man-in-the-middle attacks, causing disruption for specific nodes by isolating them, or selectively dropping packets for a particular destination. A compromised node can perform the attack by falsifying the answers to probe requests to create the perception of a route with higher utility from the perspective of the victim node. The victim will make an incorrect change since the perceived benefit does not reflect reality. For example, if the adaptation decision is based on the bandwidth from the source, a malicious node can attract other nodes to select it as parent by lying about its bandwidth when it is probed. The victim nodes will incorrectly choose to adapt and select the malicious node as parent since it appears that the change will guarantee a better bandwidth from the source. The malicious node can augment the attack by lying about other metrics such as latency or saturation.

Repulsion attacks seek to reduce the attractiveness of other nodes or misrepresent their ability, with the ultimate goal of free-loading, traffic pattern manipulation, or augmenting attraction attacks. As in the case of attraction attacks, repulsion attacks can target one particular node. One way a malicious node can conduct the attack is by lying about its performance. For example, a malicious node may lie about route costs (i.e., hop count) in order to convince other nodes that it has a bad connection and thus it should not be selected as a parent. The malicious node will then obtain a reduced burden while still taking advantage of the system.

As many nodes share the same physical links, an attacker may instead choose to manipulate the physical or logical infrastructure to affect the performance metrics monitored by a victim node by exploiting its physical connectivity to the victim. For example, a node can affect the link state estimation by injecting a very small amount of traffic for a short amount of time, creating the perception that the performance degraded significantly and convincing the victim node to change its parent. A variant of the attack is to target the active probes on which the victim node relies. In this case, the victim's peers will be made to look unappealing as possible parents, thereby increasing the chances of the malicious node moving closer to the source in the multicast structure.

Disruption attacks target the availability of the network by using the adaptation process to turn the system against itself. An attacker can create significant disruption in the overlay by injecting or influencing the observation space metric data to generate self-destructive responses as a result of unnecessary adaptations. The ultimate goal of such attacks is to affect the infrastructure that supports the overlay with the intent to prevent or degrade service. These attacks can be classified as a form of denial of service (DOS) and can result in jitter, flapping, or partitioning the overlay.

IV. Defending Against Attacks in Adaptive Overlay Networks

In this section, we describe a comprehensive solution for mitigating insider attacks that exploit adaptation in overlay networks. As the attacks we are concerned with are performed by compromised nodes controlled by adversaries, the solution space components we describe below are complementary to authentication and integrity mechanisms.

A. Solution Space

We identify four components that a framework designed to address insider attacks against adaptation must include. Due to lack of space, we provide a high-level description of all of them and a detailed description of a critical component: reducing incorrect or unnecessary adaptations. More details about each component can be found in [29].

• *(A1) Reducing incorrect adaptations:* A node makes adaptation decisions based on two types of information: the performance from the source measured directly by each node and the performance of the neighbor nodes obtained by probing them. By blindly accepting the information reported by the potentially malicious probed nodes, correct nodes may make incorrect decisions. We propose to prevent incorrect adaptations by detecting and filtering out outliers in the metrics reported by malicious nodes. Our method evaluates temporal and spatial correlations among data in the system. Although our solution is developed in the context of overlay networks, it can be used to address the more general problem of "blind acceptance" [30] of routing metrics present in many network protocols. We present this approach in detail in Section IV-B.

• *(A2) Increasing stability:* Reducing the number of unnecessary adaptations has the potential to increase the stability and

decrease the number of incorrect adaptations, while reducing the overhead. Nodes perceived as unstable will be pushed to the fringes of the tree as no other node will select them as a parent. We propose to integrate stability metrics such as the time a node was connected to his current parent, the frequency of changes, or the degree of variance in metrics into the function that drives the adaptation.

• *(A3) Detecting observable malicious behavior:* The methods proposed above may still result in some incorrect adaptations. However, the attacks exploiting adaptation are often used to further attack the multicast service, resulting in observable degradation of service and thus allowing additional detection mechanisms to be employed. Unlike (A1), which is focused on preventing incorrect adaptations, this component reacts to degradation of service resulting from the incorrect adaptation. We propose that every node uses the low-bandwidth, bidirectional unicast link that it shares with the source to provide feedback to the source about the received data. The link is also used by the source to inform member nodes about the state of the overlay structure to allow them to detect inconsistencies in the metrics reported by peers. The structural information can be trusted as it is sent by the source and protected cryptographically from modifications.

• *(A4) Isolating malicious nodes:* Without taking action against malicious nodes, the convergence of the protocol and the overall system overhead will increase as the malicious nodes continue to interfere with the system. We propose a gradual response where each node of the overlay maintains two dichotomous lists: a local suspect list generated by that node and a global black list generated by the trusted source based on suspect lists received from nodes in the network. The suspect lists allows nodes to take decisions locally, while the global list allows nodes to share information about malicious nodes in the system. While the benefits of the suspect list are obvious, the use of the black list requires further investigation as it creates opportunities for malicious nodes to black list other correct nodes and also increases link stress in the system.

B. Reducing Incorrect Adaptations Using Local Spatial and Temporal Correlation for Outlier Detection

The primary cause of the identified attacks is the ability of the attacker to influence the adaptation process by manipulating the performance metrics. We propose to detect inconsistent metrics by performing outlier analysis on the information received from probed nodes and used in the decision process. An *outlier* is a data point that is significantly different (greater than a threshold) from the rest of the data in the observation space based on a measure of distance.

The detection is performed locally by each node using spatial and temporal correlations. The *spatial outlier detection* compares the reported metrics received from each node in the set of probed nodes. The *temporal outlier detection* examines the consistency in the metrics received from an individual probed node over time. Our outlier detection does not affect the link stress in the system, as it uses the metrics already reported by nodes: latency, bandwidth and RTT. Both latency and RTT are utilized because they are highly correlated metrics collected in different manners (probed versus measured, respectively). Since TCP is used as the data transport protocol, loss rate is not considered. In order to avoid being suspected by correct nodes, a malicious node must insure that any lie it tells: (1) is consistent with what the other peers are reporting during a probe cycle about current network conditions, (2) ensures consistency between the different dependent metrics (bandwidth, latency, and RTT), and (3) is consistent with metrics it reported in the past. The spatial outlier detection targets the first and second aspects of consistency, while the temporal outlier detection targets the second and third aspects. Spatial and temporal data correlations have been previously shown effective in detecting network attack scenarios [31]. Unlike the the general approach in [31], our work does not look for correlations but exploits the fact that they exist to detect suspicious nodes.

The intuition behind our solution is that the intrinsic dependency existent in the measured variables requires attackers to make sure the "fake" metrics vary in a consistent manner. This dependency results from a fundamental characteristic of end-system multicast systems – that the distribution tree overlaps on the routing infrastructure, often represented as a measure called link stress. Lying is made more difficult by the fact that attackers can only make the RTT worse, because it is a measured attribute, and yet, at the same time, the RTT must remain consistent with both the bandwidth and latency. Our solution also forces an attacker to lie consistently with other peers. This is difficult to achieve as an attacker does not have perfect knowledge of the observation space, must accurately predict the random subset of nodes that will be queried, and only has a finite amount of time (the probe period) to coordinate with other attackers.

A key component of our approach is using the Mahalanobis [32] distance to detect outliers. We selected this distance function because it has been shown to detect outliers with multiple attributes better than other distance functions [33], scales each variable based on its standard deviation and covariance, and takes into account how the measured attributes change in relation to each other. This makes it appropriate for our environment where there is a dependency between several of the attributes reported by each node.

Spatial outlier detection. The outlier detection is performed by a node as follows. Each probe cycle, the node first computes the centroid of the data set consisting of observation tuples from all probed nodes. An *observation tuple* is represented by bandwidth, latency, and RTT. The node then computes the Mahalanobis distance between the observation tuple from each probed node and the centroid as follows:

$$d(\vec{x}, \vec{y}) = \sqrt{((\vec{x} - \vec{y})^T C^{-1} (\vec{x} - \vec{y}))} \quad (1)$$

where $\vec{x}$ and $\vec{y}$ are the feature vectors consisting of bandwidth, latency, and RTT. $\vec{x}$ is the value from the probe response and $\vec{y}$ is the average value that was calculated. C^{-1} is the inverse covariance matrix computed from the observation tuples. When there are not enough observation tuples received

during a probe cycle, the tuples are compared with the most recent centroid. When there is no variance between the received observation tuples, the Mahalanobis distance cannot be computed since the determinant of the covariance matrix becomes zero. In this case, a node is randomly selected from that probe set of observation tuples and compared to the most recent centroid. If no centroid is available, the decision is postponed to the next probe cycle.

Spatial threshold selection. The threshold for our outlier detection can be mathematically derived as in [34], [35], assuming a multivariate Gaussian distribution for the metrics vector. The contours of equal probability of this distribution create a 3-dimensional ellipsoid and the outlier threshold reflects the probability of a vector being within the ellipsoid specified by the focus k. The probability that a random vector lies within the ellipsoid increases with the size of k. Thus, for a given value of k the probability that a probed tuple lies within the ellipsoid can be computed as:

$$P = -\frac{1}{\sqrt{2\pi}} + 2\left(\frac{1}{\sqrt{2\pi}}\int_0^k e^{\frac{y^2}{2}}dy\right) - \sqrt{\frac{2}{\pi}}ke^{\frac{-k^2}{2}} \qquad (2)$$

We initially selected a k of 2.37, creating a threshold which half of the probes would successfully pass. Through testing in over 539,739 probe responses during 19,465 probe cycles, we found an ellipsoid determined by a threshold of 1.5 will contain approximately 80% of the nodes. Thus, we selected a threshold of 1.5 for our experiments. This variation from the mathematically derived value can be attributed to the fact that the used metrics do not form a perfect normalized distribution and have a smaller variance than assumed in Equation 2. A node may select smaller threshold distances for stronger security guarantees, with the drawback that it may find itself isolated due to aggressive filtering.

Temporal outlier detection. We use temporal correlations to detect inconsistencies in the performance metrics reported over time by a node. We develop models for the peers of a given node during the course of a multicast session by using incremental learning. Our technique is based on the "simplified Mahalanobis distance" presented in [32]:

$$d(x, \bar{y}) = \sum_{i=0}^{n-1}(|x_i - \bar{y_i}|/(\bar{\sigma_i} + \alpha)) \qquad (3)$$

where n is the number of metrics, three in our case (bandwidth, latency, and RTT), $\bar{\sigma_i}$ is the standard deviation, and α is a smoothing factor empirically set to .001 to help to avoid overfitting and reduce false positives [32]. We trade-off accuracy of the distance function to minimize the amount of data we must store by making the assumption that the metrics are statistically independent. As a result, each node maintains for each peer only the temporal centroid consisting of the mean, standard deviation, and sample count computed from the observation tuples received over time, and not the whole history. The centroid for each peer is incrementally updated with observations received during each probe cycle, as in [32], using the technique Knuth described in [36]. At the end of the probe cycle, the latest observation tuple for each peer is compared with the corresponding temporal centroid using the Mahalanobis distance.

Temporal threshold selection. We used a threshold of 3.0 for our temporal outlier detection, to allow each of the three features to vary within one standard deviation from their temporally developed mean. The value was chosen based on the formula of the simplified Mahalanobis distance as in [32].

Spatio-temporal outlier detection. The two outlier detection mechanisms have the potential of being more effective when used together. We combine them by using a codebook technique similar to [31]. The peer nodes are ranked according to their spatial outlier distance from the spatial centroid and traversed from the closest to the farthest node. The node that is closest to the spatial centroid that is not a spatial or temporal outlier is chosen as the new parent. If no peer is found meeting these criteria or if there are a large number of temporal outliers, no adaptation is performed during that probe cycle.

V. Experimental Results

We demonstrate through experimental results the attacks identified in Section III and our outlier detection techniques in the context of the ESM overlay multicast system. We selected ESM because of its maturity, extensive deployment, and the advanced set of adaptation techniques it employs. Our experiments show that, although ESM employs an advanced set of adaptation mechanisms, it is unable to mitigate the attacks posed by a malicious adversary. Our outlier detection was able to reduce significantly the number of malicious changes without adding to the link stress in the system.

A. Overview of ESM

ESM [1] is a multicast system mainly used for broadcasting live events such as academic conferences. We provide a high-level description below. For further details, the reader is referred to [29]. ESM forms a peer-to-peer overlay tree for distributing multicast content. A node changes its parent in the overlay to maintain and improve application performance. Both passive observation and probing are used to collect data used to make the adaptation decision. ESM uses data sampling and data smoothing to address variations in the metrics considered: available bandwidth, latency, and RTT. ESM also employs a number of combined metrics, damping, randomization, hysteresis and three utility functions to address instabilities in the observed data. The three utility functions are based on: bandwidth, latency, and a combination of bandwidth and latency. A damping factor is used to induce stability and a randomization technique is used to avoid the case where several nodes try to change to the same parent.

In order to select a new parent, a node first computes a list of potential candidates from its neighbor set. Nodes which are currently saturated, descendants, or did not respond when recently probed are not considered. If there is no utility gain, no node is selected and the process will be repeated next cycle. If several nodes are candidates, then the first candidate is selected as the new parent. The selection process uses hysteresis to generate a negative bias against nodes that have performed poorly in the past.

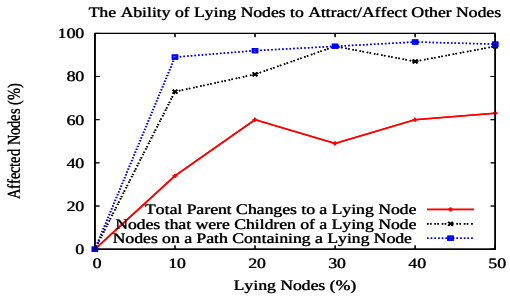

Fig. 1. The effect of attraction attacks on correct nodes for an ESM overlay of 100 nodes on PlanetLab for a duration of 60 minutes.

B. Testbed and Experiment Setup

To study the attacks and defense mechanisms under real-world conditions, we conducted our experiments on the PlanetLab [17] Internet testbed. In addition, for repulsion and disruption attacks that could have been disruptive to PlanetLab, we used DETER [18], a testbed that provides a stable, controllable emulation environment for network security research.

We use sixty minute long ESM deployments of 100 nodes in which the nodes join after the experiment begins and leave before it ends, with an average participation time of fifty-five minutes. As in previous ESM deployments [37], nodes are probed every seven seconds, the saturation degree of correct nodes is six, and the source constant bit rate is 480 Kbps. All experiments use these parameters unless otherwise noted.

C. Attack Effectiveness

1) Attraction Attacks: We demonstrate the effect that a single coalition of one malicious node, who exploits the adaptive nature of ESM, has on the multicast tree construction, maintenance, and stability. One randomly selected node performs an attraction attack in which it lies every probe cycle about having the best bandwidth (480Kbps), latency (0ms), and no saturation. We summarize our findings in Table I. When the node is honest, it is selected only 5 times as a parent by other nodes. However, when the node is malicious, it is selected 172 times, or almost 35 times more often. The malicious node causes the overlay to become more unstable, as can be seen in the large increase of total parent changes. This increased instability can be attributed to the fact that the new child will eventually realize the bait-and-switch and change parents again.

We next consider the effect on the correct nodes when a percentage of randomly selected malicious nodes perform attraction attacks. Metrics we investigate are: the percentage of nodes that have at least one malicious node on their path to the source, the percentage of nodes that have a malicious node as a parent at some point during the experiment, and the number of parent change decisions that resulted in selecting a malicious node. The results of the experiment, summarized in Fig. 1, demonstrate that even a small percentage of malicious nodes will affect the majority of correct nodes in the overlay. Fluctuations in the general trends of the curves result from the use of real-world experimentation and randomly selected malicious nodes. The greater the number of malicious nodes

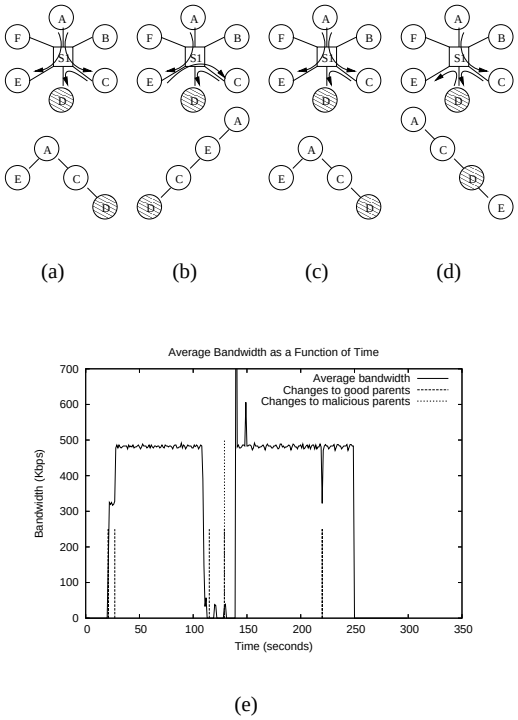

Fig. 2. An example of repulsion attack against an ESM overlay in a controlled experiment on DETER. (a) represents the overlay and the multicast tree before attack while (b), (c) and (d) are the topology changes in the multicast tree as a result of the attack. Node E is manipulated by the attacker to attach to malicious node D, although this causes E to be three hops away from the source, instead of just one. (e) The average bandwidth with topology changes denoted by the solid and dashed impulses representing good and malicious parent changes respectively.

located near the source in the overlay topology, the greater the effect will be on the overall system.

2) Repulsion Attacks: While performing experiments, we noticed that nodes with very good performance, such as those directly attached to the source, could not be fooled by malicious nodes simply by lying, and more sophisticated attacks are needed. We demonstrate a repulsion attack where an attacker affects the partially observable link state estimation in order to make a node incorrectly believe that the performance from the current parent is inadequate.

Fig. 2 presents a star topology composed of six nodes, all of which are connected with 100 Mbps links to switch S1. For demonstrative purposes, ESM is configured to use a saturation degree of two. In our example, node A is the source and nodes C, D, and E are end-systems in the overlay. Nodes B and F are outsiders who collude with D, a malicious node that has infiltrated the overlay. During the attack, nodes B and F generate traffic to augment the attack of malicious node D, which lies about its bandwidth (480Kbps), latency (0ms), and saturation (none). Similar results will be obtained if nodes B and F are trusted members of the overlay attempting to improve their position in the tree or influence the path data takes from the source to themselves or others.

The overlay initially converges to the stable structure seen in Fig. 2(a), at which point the mean bandwidth is approximately 480 Kbps. Topology changes occur at the impulses seen in

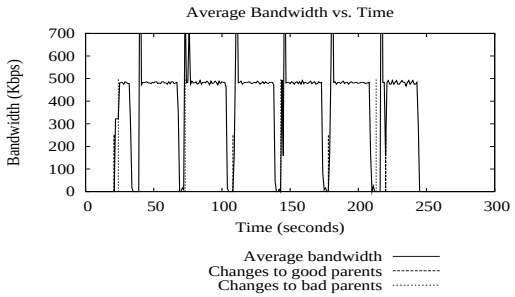

Fig. 3. An example of disruption attack against an ESM overlay in a controlled experiment on DETER. The experiment was performed using the same experimental setup as Fig. 2. The attackers periodically sent 5 second bursts of traffic at the focal point creating constant churn in the system. Topology changes are denoted by the solid and dashed impulses representing good and malicious parent changes respectively.

Fig. 2(e). The attack begins at 115 seconds when nodes B and F begin flooding 30 seconds worth of traffic at the source, node A. After several seconds of traffic, the attack is able to generate the first change in the tree when node C chooses node E as its new parent in Fig. 2(b), despite the fact that C will be an extra hop to the source. Then, 14 seconds later, C switches back to its previous position, but the overlay has yet to stabilize in Fig. 2(c). Next, node E detaches from the source and, instead of choosing node C, chooses the malicious node D, as its parent in Fig. 2(d). Note that node E was previously directly connected to the source but it is now connected three hops away. The changes after 200 seconds are due to nodes leaving the experiment.

The cost of such an attack consists of saturating the 100 Mbps link with a short 30 second burst of traffic. In real Internet deployments, the cost of the attack will be substantially less since links will typically have a lower capacity.

3) Disruption Attacks: Fig. 3 demonstrates an example of a disruption attack where the attacker exerts an artificial influence, extraneous traffic, towards a focal point of the overlay topology. The main difference from previous attacks is that the artificial influence is done periodically in order to destabilize the infrastructure. In the experiment in Fig. 3, the attacker sends 5 second bursts of traffic every 30 seconds. This is similar to the attacks performed in [15], [16] which targeted the TCP congestion control. Fig. 3 shows that using this technique the attacker can keep the system in a constant churn as it keeps trying to stabilize itself. Despite the fact that the attacker was using only 5 second bursts of traffic, parent changes occurred in the overlay at almost every probe cycle.

D. Effect of Malicious Nodes on Average Bandwidth

We studied the effect multiple malicious nodes can have on the overlay topology. Having a malicious parent can result in a severe degradation of service if the malicious parent decides to selectively drop data. In Fig. 4, we demonstrate the impact malicious nodes that use their position in the tree can exert on the bandwidth of correct nodes. The graphs plot the bandwidth averaged over all receivers as a function of time. Malicious nodes start dropping 100% of the data traffic received through the data dissemination tree fifteen minutes after they joined the overlay. We vary the percentage of malicious nodes to 10%, 30%, and 50% of the overlay size to demonstrate the performance degradation that results when more nodes behave maliciously.

We define the relative strength of a particular attack as:

$$\tau = \frac{B_{norm} - B_{adv}}{B_{norm} \times Num_{adv}} \quad (4)$$

where B_{norm} and B_{adv} represent the average throughput in the absence and presence of adversaries respectively, and Num_{adv} is the number of adversaries. Intuitively, tau represents the amount of damage an attack created in the system. The greater the performance degradation observed in the system between when the malicious nodes are passive and active (the difference between B_{norm} and B_{adv}), the higher the value of tau and the more damage an attack inflicts on the overlay.

Fig. 5 depicts tau varying over the percentage of the traffic dropped. As it can be seen, the greater the amount of data traffic a malicious node drops, the greater the effect it has on the system. The drop in the effectiveness of the attacks as the malicious nodes drop high percentages of data (100%) is due to ESM categorizing the malicious nodes as unstable links based on past experienced bandwidth and having a bias against choosing them as parents. Fig. 5 also shows the intuitive notion that the greater the number of malicious nodes, the greater effect there is on the system. It can be noted that just 10% malicious nodes have a significant effect on the average bandwidth. We believe this is because a percentage of 10% malicious nodes is enough to obtain advantageous positions in the vulnerable tree structure which has no path redundancy.

E. Effectiveness of Outlier Detection

To demonstrate the effectiveness of our outlier detection at improving the parent selection process and the stability of the system, we considered one malicious attacker and recorded the number of parent changes that took place for the duration of the experiment considering two cases, one when only the spatial outlier is used, and one when the temporal-spatial outlier is enabled. The outcome of these experiments is shown in Table I. The results indicate that using the spatial outlier detection scheme has dramatically reduced the likelihood of choosing a malicious parent since the number of times the malicious node was selected as a new parent is reduced from 172 to 70. The addition of the temporal outlier detection further reduces this to only 35 times.

Our method also dramatically improved the stability of the overlay in spite of the presence of the malicious node, as measured by the decrease in total parent changes denoted in third column of Table I. In fact, the number of adaptations is comparable to the number of adaptations that would occur when no malicious nodes are present in the overlay.

F. Coalitions of Attackers and Spatial Outlier Detection

The previous experiment demonstrated the effectiveness of the spatial correlation for detecting outliers produced by a single coalition containing one attacker. We now consider the constrained collusion model presented in Section II-B in which

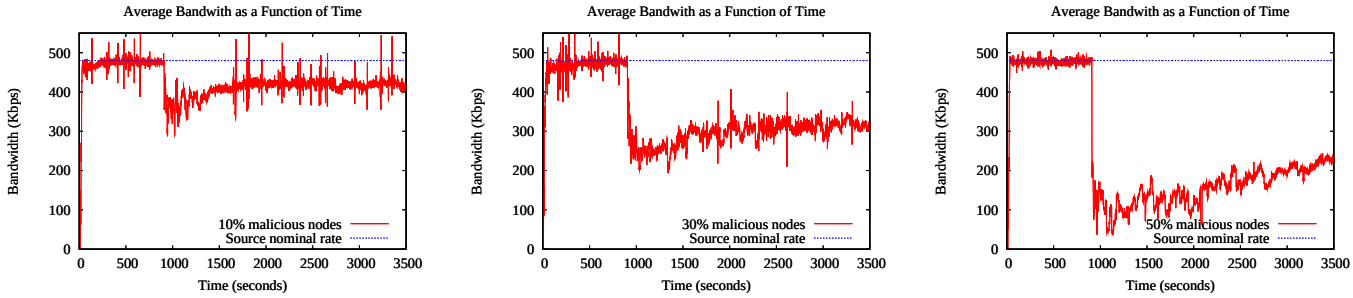

Fig. 4. The average bandwidth over time for an ESM overlay of 100 nodes on PlanetLab for a duration of 60 minutes with different percentages of malicious nodes.

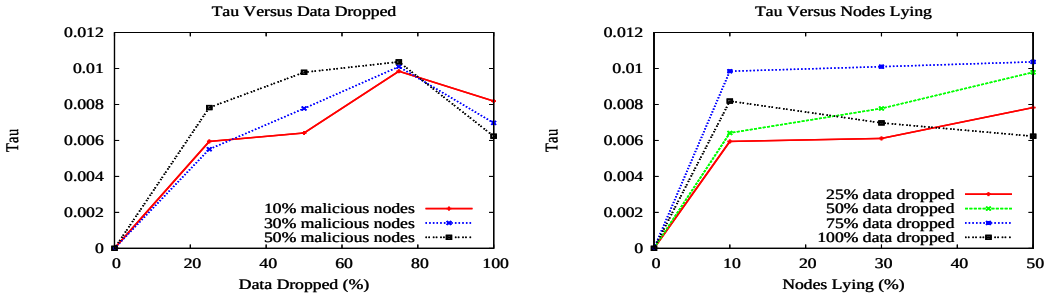

Fig. 5. Tau as a function of (a) the percentage of data dropped and (b) the percentage of malicious nodes for an ESM overlay of 100 nodes on PlanetLab for a duration of 60 minutes.

TABLE I

THE EFFECTIVENESS OF OUTLIER DETECTION AT IMPROVING PARENT SELECTION FOR AN ESM OVERLAY OF 100 NODES ON PLANETLAB OVER 60 MINUTE RUNS

Experiment	Changes to Malicious Parents	Total Parent Changes
No lying	5	833
Lying	172	1032
Spatial	70	800
Spatial/Temp	35	604

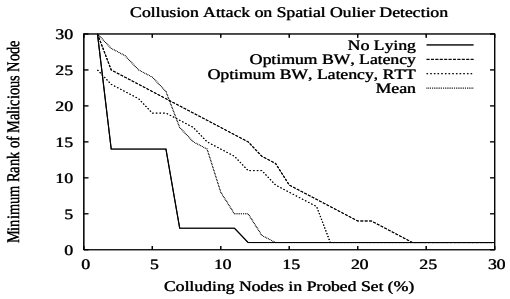

Fig. 6. Percentage of colluding nodes necessary to influence parent selection. The goal is for a colluding node to be ranked first and chosen as a parent.

all faulty nodes are part of the same coalition. A coalition of colluding attackers may attempt to bypass the outlier detection mechanism itself by shifting the centroid so they are not perceived as outliers anymore. As a result, one of the members of the malicious coalition will be selected as the parent.

We consider three colluding cases requiring different degrees of coordination between the attackers. In the first case, referred to as "Optimum BW, Latency", the malicious nodes can only lie about the latency and bandwidth and not the RTT. The second case, referred to as "Optimum BW, Latency, RTT", the malicious nodes agree to lie consistently on a set of predefined values: RTT of 0, latency of 0, and bandwidth of 480 Kbps. Note that in order to influence the RTT, this case requires that one malicious node indeed has an RTT of 0 with the victim and it can intercept all the RTTs of the other nodes in the coalition. The third case, referred to as "Mean", assumes that the attackers have the ability to share their observed performance and compute and report the average of their real metrics, again only bandwidth and latency. This case requires strong coordination between the attackers, which may not always be possible during a probe cycle without creating inconsistencies in measured probe times. We compare these cases with the normal case, when no nodes are lying.

We summarize our findings for an ESM overlay of 118 nodes on PlanetLab in Fig. 6. The graph depicts the rank of possible parents of a malicious node member of a coalition. Note that it took a malicious coalition of 80% of the nodes in a probe set in the first case and 60% of the nodes in a probe set in the second case before a malicious node is chosen as the next parent. This demonstrates the effectiveness of the spatial outlier detection since both the number and type of metrics used by the outlier detection defense make it difficult for the attackers to maintain consistency. In the "Mean" case, 47% of the nodes needed to be in a coalition before they could deterministically guarantee that a malicious node would be chosen. This demonstrates that if the attackers have more information, then they can reduce the amount of

work necessary for subverting the spatial outlier detection mechanism. When compared with the normal case in which no node exhibits malicious behavior, the "coalition" would only need to contain 40% of the nodes. Thus, lying about metrics, even with sophisticated coordination techniques, is no longer an effective attack technique. The spatial outlier technique we describe constrains the behavior of attackers and reduces their ability to artificially augment their influence on the system.

G. Overhead and System Performance

Our outlier detection does not introduce any extra link stress since it uses information that is already being exchanged between nodes. The memory utilization for spatial correlation only lasts for the span of a probe cycle and requires maintaining the observation tuple associated with each of the probed nodes, while the storage requirements consist of three additional values in the route table for the peer set maintained by each node. In the case of the temporal outlier detection, the memory usage consists of maintaining the temporal centroid. By incrementally updating the centroid, we do not need to maintain the entire history for each probed node. The temporal outlier detection also requires modifying the route table entries to store nine additional values: mean, standard deviation, and count for each of the three metrics.

VI. Related Work

Our work focuses on attacks exploiting measurement-based adaptation in overlay networks and our solution uses concepts borrowed from anomaly detection. Below we review work in several areas related to our research.

Attacks exploiting adaptivity. Previous work showed the vulnerability of the TCP adaptation mechanisms, i.e. the congestion control mechanism, to malicious attacks [15]. The authors showed that by manipulating the end-system's perception of network congestion, the adaptivity mechanism could be used to perform a low-rate DOS attack with severe effects on TCP throughput. The attack was generalized in [16], as a form of low-rate ROQ attack targeting point-to-point adaptive control loops that drive resource allocation and affect perceived service of a system (bandwidth, jitter, etc).

Our work assumes a different, stronger adversarial model in a distributed system, specifically overlay networks. The nature of the attacks, application and deployment environment allows us to use a context sensitive observation space and correlated information associated with the same information that drives the adaptation to detect and limit the effect of malicious behavior.

Anomaly detection and Mahalanobis distance. Recently the benefits of the Mahalanobis distance for statistical anomaly detection have been demonstrated in the context of network intrusion detection [32], [38]. In [38] the authors present a comparative study of detection schemes based on data mining techniques for network based intrusion detection. In [32] the authors discuss an unsupervised, payload-based network anomaly detector based on the Mahalanobis distance which was used to detect attacks like worms.

Use of spatial and temporal correlations. Spatial and temporal correlations were previously used in the context of network security. A notable work in this aspect is [31] where authors use temporal and spatial correlations to trace back attacks and detect attack scenarios, using a large amount of information from intrusion detection systems, firewalls, and different software logs. Unlike the approach in [31], which was more general, our work focuses on overlay networks and does not look for correlations, but exploits the fact that they exist to detect inconsistent metrics and find suspicious nodes.

Correlations have also been used in sensor network and ad-hoc networks for the detection of malicious nodes [39], [40]. Most of this research focused on the evaluation of off-line data developed in a simulator. In our work, the correlation is actually incorporated in-line with the protocol as it tries to adapt. Analysis is performed on the Internet with real data while fusing multiple correlations to improve our predictive abilities. The work in [40] shows how to augment a sensor network with spatio-temporal correlation to detect misinformation being injected into the sensor streams. In our research, we are concerned with an attacker manipulating the control information in order to influence system adaptation.

Malicious behavior in overlay networks. The problem of malicious attackers was previously studied in the context of structured overlay networks. A subset of these types of attacks, referred to as Eclipse attacks [13], [14], was subsequently studied in optimized structured file sharing overlays. The solution enforces degree constraint invariants associated with neighbors, supported by anonymous auditing, and takes advantage of strong organizational neighbor constraints existent in such networks. As unstructured overlay networks do not have such constraints, the proposed solutions are not applicable.

To the best of our knowledge, the problem of malicious insider attacks was not studied in the context of unstructured overlay networks. An attack performed by selfish attackers (i.e. nodes that want to obtain an advantage but do not have destructive goals) was shown through simulations in [41]. Our work is different in the fact that it considers malicious attackers and presents results in the context of a real system in real deployments over the Internet.

VII. Conclusions

In this paper we identified insider attacks that exploit measurement-based adaptation mechanisms in multicast overlay networks. We discussed a comprehensive defense framework and presented an in-depth solution to a critical aspect of the problem: preventing poor adaptation decisions in networks influenced by attackers. Our solution lies in performing spatial and temporal outlier analysis on measured and probed metrics to allow an honest node to make better use of available information before making an adaptation decision. We demonstrated the effectiveness of the newly identified attacks and the benefits of using our outlier detection and response mechanisms in the context of ESM, a well-known adaptive multicast overlay network. Our experiments conducted in real-life deployments and emulations, demonstrate that although

ESM employs an advanced set of adaptation mechanisms it is unable to mitigate the attacks posed by a malicious adversary.

Previous research has demonstrated the inability of conventional detection techniques to detect attacks on adaptive protocols. In this research, we have demonstrated the importance of tightly coupling the detection space and the control space. We showed that by incorporating context sensitive anomaly detection into the protocol, the detection mechanisms have the semantic understanding to improve the adaptive decision process. Our experiments demonstrate that our techniques improve the adaptation process and the overall stability of the system while limiting the effect of malicious nodes.

Current work investigates the trade-offs and benefits of sharing information about malicious behavior using a two-stage response mechanism relying on local and global knowledge.

VIII. Acknowledgments

We would like to thank Nick Petroni for insightful comments that significantly improved the presentation of this work, Sanjay Rao and Ruben Torres for providing insights into ESM, and Mehmet Demirci for his help with the selective dropping experiments. We also thank the anonymous reviewers for their insightful comments. This work is supported by National Science Foundation CyberTrust Award No. 0430276. The views expressed in this research are not endorsed by the National Science Foundation.

References

[1] Y. Chu, S. G. Rao, and H. Zhang, "A case for end system multicast (keynote address)," in *SIGMETRICS '00*, 2000.

[2] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," in *ACM Sigcomm*, August 2002.

[3] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel, "ALMI: An application level multicast infrastructure," in *3^{rd} USENIX Symposium on Internet Technologies and Systems (USITS)*, 2001.

[4] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O'Toole, Jr., "Overcast: Reliable multicasting with an overlay network," in *USENIX OSDI 2000*, 2000.

[5] M. Castro, M. Costa, and A. Rowstron, "Should we build Gnutella on a structured overlay?," *SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 131–136, 2004.

[6] M. Castro, P. Druschel, A. Kermarrec, and A. Rowstron, "SCRIBE: A large-scale and decentralized application-level multicast infrastructure," *IEEE Journal on Selected Areas in communications (JSAC)*, vol. 20, no. 8, pp. 1489–1499, 2002.

[7] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "Splitstream: High-bandwidth multicast in cooperative environments," 2003.

[8] "2005 e-crime watch survey - survey results." http://www.cert.org/archive/pdf/ecrimesurvey05.pdf.

[9] D. S. Wallach, "A survey of peer-to-peer security issues," in *International Symposium on Software Security*, pp. 42–57, 2002.

[10] W. J. Bolosky, J. R. Douceur, D. Ely, and M. Theimer, "Feasibility of a serverless distributed file system deployed on an existing set of desktop PCs," in *SIGMETRICS '00: Proceedings of the 2000 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, (New York, NY, USA), pp. 34–43, ACM Press, 2000.

[11] E. Sit and R. Morris, "Security considerations for peer-to-peer distributed hash tables," in *IPTPS '01*, (London, UK), pp. 261–269, Springer-Verlag, 2002.

[12] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," in *OSDI '02*, pp. 299–314, ACM Press, 2002.

[13] A. Singh, M. Castro, A. Rowstron, and P. Druschel, "Defending against eclipse attacks on overlay networks," in *Proceedings of the 11^{th} ACM SIGOPS European Workshop*, (Leuven, Belgium), September 2004.

[14] A. Singh, T.-W. Ngan, P. Druschel, and D. Wallach, "Eclipse attacks on overlay networks: Threats and defenses," in *The 25^{th} Conference on Computer Communications*, (Barcelona, Spain), April 2006.

[15] A. Kuzmanovic and E. W. Knightly, "Low-rate TCP-targeted DOS attacks: the shrew vs. the mice and elephants," in *SIGCOMM '03*, 2003.

[16] M. Guirguis, A. Bestavros, and I. Matta, "Exploiting the transients of adaptation for RoQ attacks on internet resources," in *The 12^{th} IEEE International Conference on Network Protocols (ICNP'04)*, 2004.

[17] "Planetlab." http://www.planet-lab.org/.

[18] "Deter." http://www.isi.edu/deter/.

[19] J. F. Nash, "The Bargain Problem," *Econometrica*, vol. 18, pp. 155–162, 1950.

[20] D. G. Andersen, "Resilient overlay networks," Master's thesis, Department of EECS, MIT, May 2001. http://nms.lcs.mit.edu/projects/ron/.

[21] D. Bauer, S. Rooney, P. Scotton, S. Buchegger, and I. Iliadis, "The performance of measurement-based overlay networks," in *QofIS*, pp. 115–124, 2002.

[22] Y. Chu, S. G. Rao, S. Seshan, and H. Zhang, "Enabling conferencing applications on the internet using an overlay multicast architecture," in *ACM SIGCOMM 2001*, (San Diago, CA), ACM, Aug. 2001.

[23] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP 4)." Internet Engineering Task Force: RFC 1771, March 1995.

[24] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed internet routing convergence," *IEEE/ACM Trans. Netw.*, vol. 9, no. 3, pp. 293–306, 2001.

[25] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz, "Route flap damping exacerbates internet routing convergence," in *SIGCOMM*, pp. 221–233, 2002.

[26] M. Seshadri and R. H. Katz, "Dynamics of simultaneous overlay network routing," Tech. Rep. UCB//CSD-03-1291, University of California, Berkeley, November 2003.

[27] C. Tang and C. Ward, "GoCast: Gossip-enhanced overlay multicast for fast and dependable group communication," in *DSN '05*, pp. 140–149, IEEE Computer Society, 2005.

[28] B. Y. Zhao, L. Huang, J. D. Kubiatowicz, and A. D. Joseph, "Exploiting routing redundancy using a wide-area overlay," Tech. Rep. CSD-02-1215, U. C. Berkeley, Nov 2002.

[29] A. Walters, "Mitigating attacks against adaptation mechanisms in overlay networks," Master's thesis, Purdue University, May 2006.

[30] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, "Detection of invalid routing announcement in the internet," in *DSN '02*, pp. 59–68, IEEE Computer Society, 2002.

[31] G. Jiang and G. Cybenko, "Temporal and spatial distributed event correlation for network security," in *American Control Conference*, 2004.

[32] K. Wang and S. J. Stolfo, "Anomalous Payload-based Network Intrusion Detection," in *Proceedings of the Recent Advances in Intrusion Detection (RAID) Conference*, September 2004.

[33] C. Lu, D. Chen, and Y. Kou, "Multivariate spatial outlier detection," *International Journal on Artificial Intelligence Tools, World Scientific*, vol. 13, pp. 801–812, December 2004.

[34] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty," *International Journal of Robotics Research*, vol. 5, no. 4, pp. 56–68, 1986.

[35] M. I. Ribeiro, "Gaussian probability density functions: Properties and error characterization," 2003.

[36] D. E. Knuth, *The Art of Computer Programming, 2nd Ed. (Addison-Wesley Series in Computer Science and Information*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1978.

[37] Y. Chu, A. Ganjam, T. E. Ng, S. G. Rao, K. Sripanidkulchai, J. Zhan, and H. Zhang, "Early experience with an internet broadcast system based on overlay multicast," in *USENIX Annual Technical Conference, General Track*, pp. 155–170, 2004.

[38] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," in *Proceedings of the Third SIAM International Conference on Data Mining*, 2003.

[39] Y. an Huang, W. Fan, W. Lee, and P. S. Yu, "Cross-feature analysis for detecting ad-hoc routing anomalies," in *ICDCS '03*, (Washington, DC, USA), p. 478, IEEE Computer Society, 2003.

[40] S. Tanachaiwiwat and A. Helmy, "Correlation analysis for alleviating effects of inserted data in wireless sensor networks," in *MobiQuitous*, pp. 97–108, 2005.

[41] L. Mathy, N. Blundell, V. Roca, and A. El-Sayed, "Impact of simple cheating in application-level multicast," in *INFOCOM*, 2004.