Information Visualization (2007) 6, 3–17

www.palgrave-journals.com/ivs

Interactive wormhole detection and evaluation

Weichao Wang[1]
Aidong Lu[2]

[1]Department of EECS, University of Kansas, Lawrence, KS, U.S.A.; [2]Computer Science Department, University of North Carolina at Charlotte, Charlotte, NC, U.S.A.

Correspondence:
Aidong Lu, Computer Science Department, University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, NC, 28223, U.S.A.
E-mail: alu1@uncc.edu

Abstract
Wormhole attacks in wireless networks can severely deteriorate network performance and compromise security through spoiling the routing protocols and weakening the security enhancements. This paper develops an approach, interactive visualization of wormholes (IVoW), to monitor and detect such attacks in large-scale wireless networks in real time. We characterize the topology features of a network under wormhole attacks through the node position changes and visualize the information at dynamically adjusted scales. We integrate an automatic detection algorithm with appropriate user interactions to handle complicated scenarios that include a large number of moving nodes and multiple wormhole attackers. Various visual forms have been adopted to assist in the understanding and analysis of reconstructed network topology and to improve the detection accuracy. Extended simulation has demonstrated that the proposed approach can effectively locate the fake neighbor connections without introducing many false alarms. IVoW does not require the wireless nodes to be equipped with any special hardware, thus avoiding any additional cost. We have performed user studies to evaluate the effectiveness of our approach and demonstrate that visual analysis can be successfully combined with network security mechanisms to greatly improve intrusion detection capabilities.
Information Visualization (2007) **6,** 3–17. doi:10.1057/palgrave.ivs.9500144

Keywords: Interactive detection; wormhole attacks; visualization on network security; wireless networks; topology visualization

Introduction

The intrusion detection system (IDS) in wireless networks[1] has played an important role in network security by providing an additional level of protection to the network topology and applications beyond the traditional security mechanisms such as encryption and authentication. It detects attacks and isolates malicious nodes by matching the patterns of known intrusions or discovering anomalies[2–5] in network activities. Its application environments cover almost all wireless networking scenarios such as *ad hoc* networks,[1] wireless LANs,[6] and sensor networks.[2,3] A good survey can be found in Zhang *et al*.[7]

With the fast increases in data scale, available bandwidth and protocol diversity in wireless networks, intrusion detection mechanisms must uncover the patterns of known attacks or the anomalies caused by unknown intrusions from a continuous, multivariate data flow in real time. Therefore, effective representation of the data is essential for users to understand the hidden information and for IDS to preserve detection accuracy and efficiency. Visualization techniques, which enable the derivation of insights from massive and dynamic data, provide a powerful tool to satisfy these requirements. In addition to information representation, visualization techniques also provide highly interactive interfaces to accelerate visual analytics. There has been pioneer research done on visualization

Received: 23 June 2006
Revised: 31 July 2006
Accepted: 21 October 2006
Online publication date: 25 January 2007

for computer security. The adopted methods include multi-resolution data details,[8] visual correlation among different parts of the data,[9,10] and time-varying patterns.[11,12]

Most of the existing IDS approaches depend on the measurements of some network parameters (e.g. packet delivery ratio, end-to-end delay) to identify the attacks. Therefore, the detection capabilities will be restricted by acquirement delay and accuracy of these measurements. Since many attacks on wireless networks target the network topology (e.g. neighbor discovery and routing), new approaches are expected to detect such attacks based on more direct 'evidences.'

In this paper, we explore the development of approaches that can detect attacks on wireless networks directly based on their impacts on the network topology. To demonstrate the proposed method, we choose a specific attack, the wormhole attack,[13–15] as our research problem. In a wormhole attack, malicious nodes will tunnel the eavesdropped packets to a remote position in the network and retransmit them to generate fake neighbor connections, thus spoiling the routing protocols and compromising some security mechanisms. The impacts of wormhole attacks on the reconstructed topology of a 2D wireless network are illustrated in Figure 1. The simulation results in Hu and Evans[16] and Kong *et al.*[17] have shown that when there are more than two wormholes in the network, more than 50% of the data packets will be attracted to the fake neighbor connections and get discarded.

A preliminary approach, MDS-VoW, to wormhole detection using visualization techniques was proposed by Wang and Bhargava.[18] This approach uses multidimensional scaling (MDS) to reconstruct the topology of a wireless network and locates the wormholes and fake neighbor connections by identifying distortions in the reconstruction result. Although effective as a proof-of-concept prototype, MDS-VoW has several deficiencies when it is applied to real wireless environments. First, the authors only evaluate MDS-VoW in a network containing a few hundred nodes. Its performance and detection accuracy in a larger scale environment (e.g. thousands of nodes) remain undetermined. Second, MDS-VoW assumes that all nodes are static. Therefore, its detection capability in mobile wireless networks has not been investigated. Finally, the experimental results focus on the scenarios when only one wormhole exists in the network while the research in Hu and Evans[16] and Kong *et al.*[17] has demonstrated that multiple wormholes put more severe impacts on the network performance.

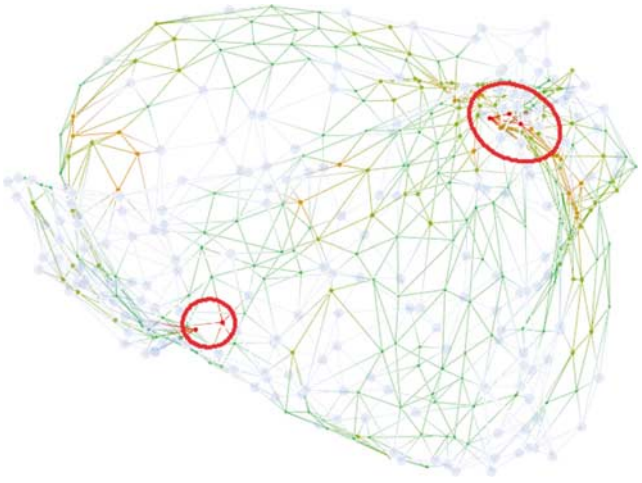

Figure 1 Visualization for wormhole detection. The two red circles indicate suspicious regions. We have combined interactive visualization into the wormhole monitoring, representation and detection processes to analyze the potential wormhole attack regions in a large-scale wireless network.

The method introduced in this paper, interactive visualization of wormholes (IVoW), provides a visual approach through which users can detect multiple wormholes in a large-scale, mobile wireless network. It first reconstructs network topology based on the measured distances among neighboring nodes. To reduce the network reconstruction overhead caused by node movement, incremental MDS[19,20] is adopted. Adaptable representation of the reconstruction result with attack-dependent level-of-detail will assist users to identify the 'suspicious areas' under wormhole attack. Multiple rounds of detection with false-alarm reduction methods are developed to improve the detection accuracy when multiple wormholes coexist in the system.

As we demonstrate, the proposed visualization approach can effectively identify the fake neighbor connections. The contributions of this research can be summarized as follows: (1) We characterize the topology features of the network under wormhole attacks and present a real-time visualization approach to effectively visualize and monitor topology changes. (2) We integrate interactive visualization into multiple steps of intrusion detection procedures, including representation, monitoring and detection. This approach significantly accelerates the detection procedure by taking advantage of the visual analysis capabilities of human experts. (3) IVoW directly uses the topology information to detect attacks on wireless networks, thus avoiding the overhead and inaccuracy caused by the network measurements. (4) The proposed approach does not depend on any special hardware, thus avoiding any additional deployment cost.

The remainder of the paper is organized as follows: the next section provides the background of wireless networking, how wormhole attacks are conducted, and the research challenges. Then the previous research efforts that contribute to our approach are reviewed and an overview of IVoW is described. Three principal components of the proposed mechanism, efficient network reconstruction, adaptive visualization and interactive wormhole detection, are described in detail in further sections, respectively. Then the experimental results that demonstrate the improvements of IVoW over MDS-VoW are presented. We describe our user study design and analysis for evaluating

the effectiveness of our approach in the penultimate section. Finally, the last section concludes the paper and discusses future extensions.

Background

In a wireless network, the nodes communicate with each other through radio transmissions. A simplified model to describe the connectivity among wireless nodes is the unit disk graph[21]: a pair of nodes *u* and *v* can directly communicate with each other if the Euclidean distance between them is shorter than *r*, where *r* is defined as the communication range. Since the neighbor relations among wireless nodes may change because of various reasons such as node movement, device malfunction, battery exhaustion and unreliable transmission medium, a node must be able to detect its active neighbors dynamically. A widely adopted approach is to let every mobile node periodically broadcast a short message containing its identity (called 'beacon' packet), and the neighbors receiving this packet will add the node into the neighbor lists. The awareness of localized network topology and route choices is usually based on the correct establishment of and updates to the neighbor list.

The features of wireless communication enable the malicious nodes to conduct wormhole attacks. As shown in Figure 2, when a legitimate node *u* in the network sends out a beacon, the malicious node M1 can use its antenna to eavesdrop the packet, and tunnel it through a dedicated long range channel to its colluder M2. When M2 retransmits the beacon, another legitimate node *v* will receive this packet and add *u* into its neighbor list. Fake neighbor connections are generated through wormholes. Later, when data packets need to go through the wormhole, the malicious nodes may choose to discard them. Therefore, a wormhole fabricates a fake connection between *u* and *v* that is under the control of the attackers.

Wormhole attacks are difficult to detect since the malicious nodes only eavesdrop and retransmit the beacons. The adoption of a stronger encryption or authentication method will not solve the problem since the attackers act as the clone of the legitimate nodes. At the same time, the fake 'short' path between *u* and *v* may attract many data packets from their neighbors, thus deteriorating the delivery ratio and network performance.

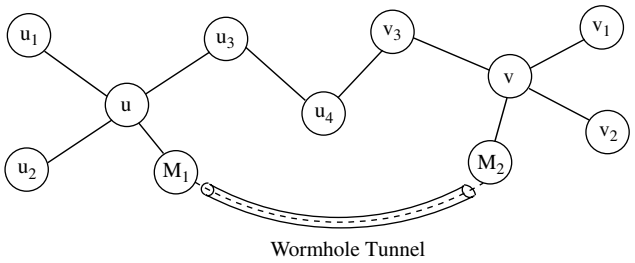

Figure 2 Wormhole attack between two nodes *u* and *v*.

Therefore, new approaches must be developed to stop these attacks.

Related work

MDS and its applications in wireless networks

Multi-dimensional scaling was originally a technique developed in behavioral and social sciences for studying the relationships among objects. The inputs to MDS are the measures of the difference or similarity between object pairs.[22] The output of MDS is a layout of the objects in a low-dimensional space. In this paper, the input is the distance matrix between the wireless nodes. The mechanism can reconstruct the network and calculate a virtual position for every node. We adopt the classical metric MDS in the proposed mechanism since the distances are measured in a Euclidean space. More details of MDS can be found in Davison[22] and Torgeson.[23]

MDS has been adopted to solve the localization and positioning problems in wireless networks. In Shang *et al.*,[24] a solution using classical metric MDS is proposed to achieve localization from mere connectivity information. The algorithm is more robust to measurement errors and requires fewer anchor nodes than previous approaches. A distributed mechanism for sensor positioning using MDS has been presented in Ji and Zha.[25] It develops a multivariate optimization-based iterative algorithm to calculate the positions of the sensors. Another approach[26] to sensor network localization adopts semi-definite programming relaxation to minimize the errors for fitting the distance measurements.

Wormhole detection in wireless networks

Wormhole attacks on mobile wireless networks were independently discovered in Dahill *et al.*,[13] Hu *et al.*[14] and Papadimitratos and Haas.[15] Below, we describe several approaches that have been developed to defend against such attacks.

If the wireless nodes are equipped with directional antennas,[16] a pair of nodes can examine the directions of the received signals from each other and a shared third node to confirm the neighbor relation. In Hu *et al.*,[14] extra information is added into a packet to restrict its transmission distance. In geographical leashes, the location information and loosely synchronized clocks together verify the neighbor relation. In temporal leashes, the packet transmission distance is calculated based on the propagation delay and signal transmission speed. In addition to the approach using scientific visualization,[18] a wormhole prevention mechanism based on graph theory is proposed in Poovendran and Lazas.[27] Using the geometric random graphs induced by the communication range constraint of the nodes, the researchers present the necessary and sufficient conditions for detecting and defending against wormholes. They also present a defense mechanism based on local broadcast keys.

Distance estimation among wireless nodes

Since MDS uses the measured distances among wireless nodes that can hear each other as inputs to reconstruct the network, we briefly introduce several distance estimation methods. The existing solutions include using received signal strength,[28] Time-of-Arrival and Time Difference of Arrival,[29,30] and triangulation.[31]

One point that we must clarify is that the measured distances cannot be directly used to prevent wormholes. For example, if the received signal strength is used to estimate the distance, the receiver cannot distinguish the resent packet by the malicious node from the real beacon. Similarly, if the nodes use the propagation delay of acoustic signals to measure the distance, the malicious nodes can easily hide the tunneling delay if radio transmission is used in the wormhole.

Visualization for computer security

With the fast development of computer security mechanisms, the scale and complexity of the security data put ever-increasing challenges to the representation and understanding of the information. Visualization techniques have been adopted by the researchers to bridge the gap. For example, it is usually difficult for the system administrators to read a text-based log file recording the traffic patterns and anomalies that happened in the past 24 h. The researchers have developed mechanisms that can provide an overview of the traffic patterns of thousands of hosts.[32] The latest approaches provide a more scalable representation capability that can cover multiple class-B IP address ranges and the intrusion alarms in them.[8,33,34]

Network scans are probably the most common and versatile intrusions. Researchers have developed a visualization methodology to characterize the scans based on their patterns and wavelet scalograms.[12] Another approach uses IP address and port number histographs to detect and analyze the scan attacks.[10] VisFlowConnect-IP[35] achieves detection of anomalous traffic through a link-based network flow visualization tool.

Under many conditions, the security data acquired from different methods must be investigated jointly to improve the detection accuracy and efficiency. The research efforts in Fink *et al*.[9] provide a visual correlation between the host processes and network traffic. In both Ren *et al*.[10] and Muelder *et al*.,[12] the approaches can identify the similarity among different scan attacks or NetFlow signatures.

While many visualization approaches to network security require large amounts of finely detailed, high-dimensional data, several mechanisms focus on the big picture. For example, the mechanism in McPherson *et al*.[36] takes very coarsely detailed data to help uncover interesting security events. The mechanism in Rafiei and Curial[37] overcomes the scalability issues inherent in visualizing massive networks through sampling. In Goodall *et al*.,[11] low level textual data are provided in the context of a high-level, aggregated graphical display. Disparate logs are also visualized to show the correlation of network alerts based on what, when, and where.[38]

Overview

An overview of working procedure of IVoW is illustrated in Figure 3. After deployment, a wireless node will estimate the distances to the other nodes that it can hear and send the measurement results to IVoW. Some fake neighbor connections through wormholes may be included. Encryption and authentication methods can be adopted to protect the integrity and authenticity of the information and prevent impersonation.

The proposed approach will use the measurement results to build the distance matrix among the wireless

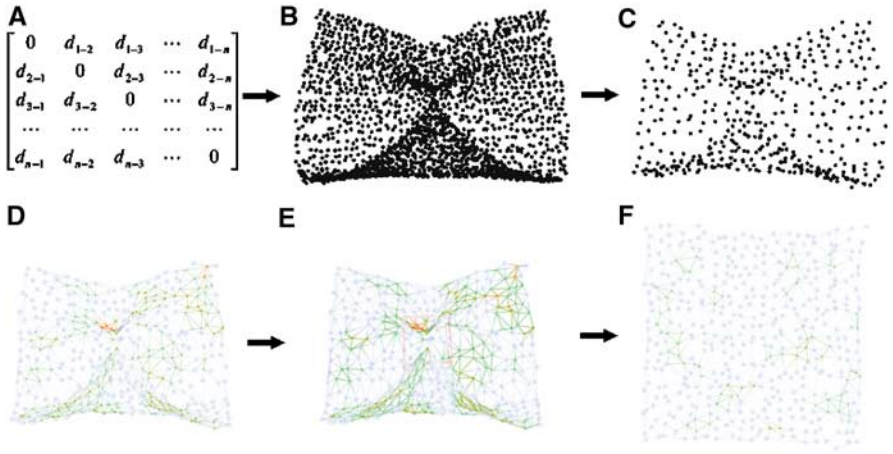

Figure 3 System overview of IVoW: A distance matrix (A) acquired from a large-scale wireless network is used to reconstruct the 3D positions of the nodes using the incremental MDS method (B), and modified through our feature point sampling (C), feature line selection (D), primitive assignments (D), and interactive detection (E) to defend against the wormhole attacks (F).

nodes and reconstruct the network topology using incremental MDS. A normalized wormhole indicator value will be calculated for every node to identify those 'suspicious areas' under wormhole attack (see the next section).

When the scale of the network and number of nodes are considered, the user may be overwhelmed by the information in the visualization. We integrate the feature element selection and attribute assignment methods to develop an adaptive visualization method. Only a part of carefully chosen nodes and their neighbor relations are illustrated while the network topology is preserved so that the suspicious areas under attack can be easily located (in section, Adaptive network visualization).

The proposed mechanism takes advantage of the users' expertise to accelerate the wormhole detection procedure and improve the detection accuracy. A set of interaction interfaces are designed to allow the users to identify the suspicious areas and activate more effective but complicated detection methods. Therefore, interactive visualization not only helps improve information understanding but also assists problem solving through visual analysis (in section, Interactive wormhole detection).

Efficient network reconstruction

Fast network reconstruction

The proposed mechanism uses MDS to reconstruct the network topology. First, every pair of nodes that can hear each other will estimate the distance between them and report it to IVoW. IVoW will calculate the average value and put the result at the suitable positions in the distance matrix. After that, a classical shortest-path algorithm, such as Dijkstra's algorithm,[39] can be used to calculate the shortest distance between every node pair. Using the distance matrix, MDS can rebuild the network layout and a virtual position for every node will be generated.

The computation complexity of traditional MDS is n^3 when there are n nodes in the network. If IVoW reconstructs the whole network from scratch after every neighbor relation change, the computation overhead will become overwhelming when n is large, thus impacting the scalability and efficiency of the proposed mechanism. To achieve efficient network reconstruction, we adopt the incremental MDS proposed in Basalaj,[19] Williams and Munzner,[20] Chalmers[40] and Morrison *et al.*[41]

The fast network reconstruction method is based on Chalmers[40] and Morrison *et al.*,[41] for which the computation complexity is n^2. Since the distances among wireless nodes seldom experience radical changes, the reconstruction result of the previous round is a good initial layout of the nodes. Single scaling will then be executed for the nodes for which neighbor relations change. The final step will include several MDS iterations upon the entire node set to refine their positions. An example of incremental MDS is illustrated in Figure 4.

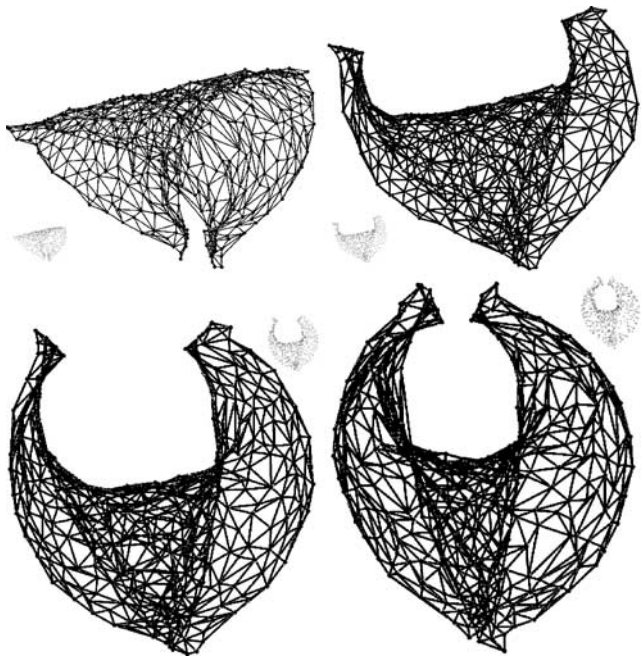

Figure 4 Example of incremental MDS: a pair of nodes slowly move to each other, thus leading to changes in network topology.

Estimating wormhole indicator value for wireless nodes

The analysis in Wang and Bhargava[18] has shown that the wormholes can be viewed as an extra force that will lead to the distortions: the distances and angles among the neighboring nodes in the reconstructed network will be very different from the values in the real layout. Subsequent research[17] has shown that the distortions in angles can be used to locate the fake neighbor connections.

For every angle formed by three neighboring nodes u_1, u_2, and v with v as the vertex, two values can be determined: (1) $\theta_{M-u_1vu_2}$, which can be calculated based on the measured distances among them; (2) $\theta_{R-u_1vu_2}$, which can be acquired from the reconstructed network. The distortions in angles can be measured by the difference between these two values.

We define a new variable, wormhole indicator (wi), for every node v based on the differences in angles:

$$wormhole\_indicator(v) = \frac{\sum \theta_{diff-u_ivu_j}}{q(q-1)},$$
$$(i, j = 1 \ldots q, i \neq j)$$

$$\theta_{diff-u_ivu_j} = \begin{cases} 0 & \text{if } \|\theta_{M-u_ivu_j} - \theta_{R-u_ivu_j}\| \leq \theta_{th}, \\ 1 & \text{if } \|\theta_{M-u_ivu_j} - \theta_{R-u_ivu_j}\| > \theta_{th}, \end{cases} \quad (1)$$

where v, u_i and u_j are neighbors, and q is the degree of connectivity of v. From the definition we find that the wormhole indicator is a normalized variable with the value range [0,1].

θ_{th} in Eq. (1) represents the threshold that is used to distinguish the changes in angles caused by the distance

measurement inaccuracy from the distortions caused by the wormholes. We adopt a format of $c(d_{err}/0.5r)$ for θ_{th}, in which d_{err} represents the distance measurement inaccuracy, r is the communication range, and c is a constant. When the distance estimation errors are not large, $d_{err}/0.5r$ roughly describes the change in angles caused by the inaccuracy. Our simulation shows that a value not smaller than 4 should be assigned to c to preserve the detection accuracy.

While the wormhole indicator values measure the impacts of the wormholes in a single network reconstruction, we also take the time factor into consideration by monitoring the distance changes among the node pairs in different reconstructions. For example, if a pair of nodes were far away from each other in the previous reconstruction and suddenly were to become neighbors, the link between them would be examined carefully to prevent wormholes.

Adaptive network visualization

With the reconstructed topology and initial wormhole indicator values, we can visualize a wireless network to monitor and detect wormhole attacks. In this section, we describe our feature element selection and attribute assignment methods to effective visualization of a large-scale wireless network.

Intuitively, we use points to represent wireless nodes, lines to strengthen the network topology, and rendering settings to reveal the intrusion detection information. This point-and-line-based visualization method is developed to satisfy the real-time rendering requirement and to provide an effective framework for illustrating network topology and security information.

Choosing a suitable resolution is one major problem for visualizing a large-scale network topology. It is difficult to observe any abnormality when there are a large number of points and lines overlapping on the screen. Multiple rendering resolutions can be used to alleviate this problem. However, it is not practical for users to adjust the suitable resolution manually, since the interaction would be too tedious for a real-time network monitoring task. Therefore, we propose to develop a self-adapted visualization method to automatically select sample points and lines.

Feature points selection

We select feature points based on their wormhole indicator values and location information to reduce the overlapping issue and preserve major topology features. Next, we discuss our ideal point distance measurement based on wormhole indicator values, the location information calculation and our feature point selection procedure.

Since the wormhole indicator is one significant feature for monitoring and detecting network attacks, we use this value to adjust the ideal point distance for each node. We draw all the points whose indicator values are larger than the threshold δ_{wi}, which is defined in the previous work[18] and can be adjusted by users, and keep a large point distance D_l for points with low indicator values. This results in a low point density on smoother surfaces and more detailed changes for abnormal regions. Practically, we use 5% of the rendering space width as D_l.

$$dis(v) = \begin{cases} D_l \times \left(1 - \dfrac{wi(v)}{\delta_{wi}}\right) & \text{if } wi(v) < \delta_{wi}, \\ 0 & \text{if } wi(v) \geq \delta_{wi}. \end{cases} \quad (2)$$

We also include the location information in the selection procedure for preserving the network topology shape, including a boundary indicator and a surface roughness measurement. We adopt the approach proposed by Rao *et al.*[42] to identify the boundary nodes of the network and assign their boundary indicators to '1'. We intend to select these points since they are important to represent the shape of the entire network. For each node v in the reconstructed topology, its normal direction $\vec{v}$ can be calculated using the best fitted plane within a local region.[18] If the set of neighbors of v is represented as N_v, the surface roughness value can be calculated using the average normal direction changes, as

$$rough(v) = \sum_{u \in N_v} \frac{(1 - \vec{v} \cdot \vec{u})}{2q},$$

where q is the number of neighbors in N_v. We intend to keep higher point density for rough surfaces, since they are more likely to contain abnormal information.

The feature value of a node is calculated as the weighted sum of the three factors: wormhole indicator, boundary indicator and roughness value. We keep the sum of the weights $w_{wi} + w_b + w_r = 1$ for saving the normalization process and we use 0.5, 0.3, 0.2 for w_{wi}, w_b and w_r, respectively, favoring the wormhole indicator values.

$$Feature(v) = w_{wi} wi(v) + w_b bound(v) + w_r rough(v). \quad (3)$$

The procedure to select feature points can be viewed as choosing a point subset that approximates the ideal point distances and achieves the maximum feature sum value. Since a greedy algorithm produces very similar results in representing the topology information, we adopt a fast algorithm by using the following selecting and updating phases.

A node with the maximum feature value is first selected through traversing the point list and added to the feature point set. Then, we update feature values of all the local points by a factor according to distance d to the selected point using $\overline{Feature}(v) = f(v) \times Feature(v)$, where

$$f(v) = \begin{cases} 1 & \text{if } d \geq dis(v), \\ \left(\dfrac{d}{dis(v)}\right)^2 & \text{if } d < dis(v). \end{cases} \quad (4)$$

We repeat this process until all the remaining points have been modified at least by a factor of $f(dis(v)/2)$. This ensures the completeness of the network topology.

We can further accelerate this selection process by selecting multiple points at each time. For the points with

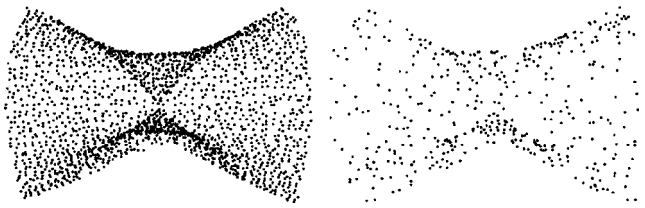

Figure 5 Point density is reduced to a smaller scale for clarity through feature point selection.

feature values larger than $\delta_{wi}/2$, we randomly select multiple points into the feature set. We can also directly use the point distance from the original distance matrix in the previous section to accelerate this process. As shown in Figure 5, the point density is reduced to a smaller scale for clarity.

Feature lines selection

We use feature lines to further strengthen the topology information by connecting selected feature point pairs. Since a wireless network usually forms a highly connected topology, we cannot illustrate every neighbor pair because of the intersecting issue. Instead, we select a small number of lines that can be used to enhance the major topology features.

To generate a succinct line drawing, we summarize three criteria for selecting feature lines.

- *Intersection*: Any two lines should not intersect on a smooth surface.
- *Connectivity*: At least one line is connected to each feature point.
- *Cell areas*: Small areas composed by the surrounding lines should be avoided for better representation.

Under these three criteria, we choose feature lines by using the Delaunay triangulation algorithm.[43] Our first two criteria are satisfied automatically and the third criterion can be approximated from the Delaunay triangulation, since it maximizes the minimum angle of all the triangles. The result of the 3D Delaunay triangulation method is used to select the feature lines between the corresponding point pairs. As shown in Figure 6, the selected feature lines enhance the main surface information in the network topology.

Attributes assignment

To effectively visualize the network features, we assign rendering attributes, including size, color and transparency, for the selected feature points and lines according to the wireless node properties.

Point size is adjusted to represent the local point density. Since the feature point selection process changes the original point density, we use the ideal feature point distance to approximate real point density, which is decided from the wormhole indicator value.

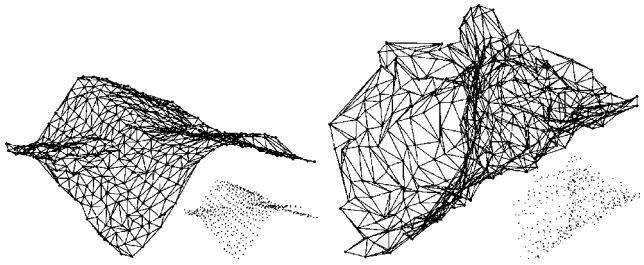

Figure 6 The automatically selected feature lines can significantly enhance the visualized network topology.

Here $\max_{size}$ represents the maximum point size in the rendering and we use 10 in our system.

$$size(v) = \begin{cases} 1 + \max_{size} \times \left(1 - \dfrac{wi(v)}{\delta_{wi}}\right) & \text{if } wi(v) < \delta_{wi} \\ 1 & \text{if } wi(v) \geq \delta_{wi} \end{cases} \quad (5)$$

Therefore, larger points represent smoother surfaces in the visualization; while smaller points indicate more abrupt changes in the topology.

The point color is assigned from blue to red to reveal the wormhole indicator value from low to high.

$$color(v) = wi(v) * C_{red} + (1 - wi(v)) * C_{blue}. \quad (6)$$

The point transparency is also calculated from its indicator as $wi(v)^{p_t}$, since the users are most interested in the potential attacked regions. P_t is set as 1.5 in our implementation. The transparent points also allow users to see through a complex topology, as shown in Figure 7.

For the situations where color is not available, a variation of Eq. (6) can be adopted to calculate the point size using a linear function between the minimum and maximum point sizes, with large points indicating the potential attackers. We still use the point transparency as above to increase the see-through effect. Figure 8 shows the two visualizations of the same data in Figure 7.

The line attributes are simply adjusted according to the attributes of the neighbor points, since we mainly use their positions to suggest the network topology. Their color and transparency are interpolated linearly between the two connecting points. We use the same thin line width to render all the lines for the least overlapping.

Interactive wormhole detection

We propose to integrate interactive visualization with intrusion detection algorithms to accelerate the detection process and improve the algorithm accuracy. Our previous work achieves a high success ratio at detecting wormhole attacks in an experimental environment. Problems occur when we apply the mechanism to networks at a larger

Figure 7 The primitive attributes are adjusted for effectively visualizing the node properties. Red color suggests potential attack regions.

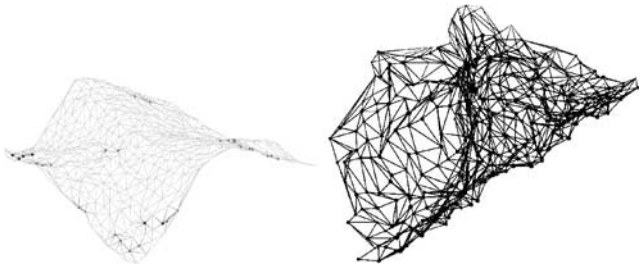

Figure 8 In black-and-white situations, node sizes are adjusted to indicate the potential attackers.

scale. As shown in Figure 14(a), the detection accuracy can decrease drastically with an increasing environmental complexity caused by multiple attackers. Therefore, we develop an interactive visualization system to handle these large, complex wireless environments. The following describes our interface design and an interactive detection procedure for wormhole detections.

Interface design

We combine the tasks of monitoring and detecting wormhole attacks into one unique system interface. Our basic idea is to visualize the network topology and potential attacks in a manner that is convenient for users to associate all the relevant information.

As shown in Figure 9, our interface is composed of three windows: topology window (bottom middle), target window (right) and history window (top left). The topology window visualizes the current network topology, where users can interact with the topology with several routine tasks, including zooming, rotating and selecting region-of-interest. The target window lists the nodes with wormhole indicator values larger than the threshold δ_{wi}. We also collect the information of each node for analysis, such as neighbor relationships, traffic history, etc. On the top left, we arrange a history window that illustrates the network topologies of previous time steps for observing the topology changes.

Interactive detection

To handle a large, complex network environment, we need to integrate user interaction with our wormhole detection and visualization methods. Our approach is to use the user inputs to guide the automatic detection procedure for further analysis. This allows the users to achieve a high success ratio with only a limited amount of simple interactions.

During the detection, the users are only required to draw a cube roughly around their region-of-interest. This cube is located by the left, back, top corner and the right, front, bottom corner. These two 3D points can be specified through the combined inputs of mouse movement and hot keys. Generally, only several operations are required to achieve a satisfying detection result.

Once the region-of-interest is located, the system will automatically process the interaction information, provide detailed analysis results, and use this information to update the network topology for further detection. Let us define the nodes within the region-of-interest as candidate nodes. Since these user-selected candidate nodes may indicate the existence of wormholes, we propose a progressive procedure to analyze them automatically.

First, we reconstruct the network topology without all the candidate nodes and calculate the stress value s_1 of MDS.

Second, we sort all the candidate nodes in a decreasing order based on their numbers of neighbors. A node with the maximum neighbor number will be added back into the network and we use incremental MDS to fast reconstruct a new topology, which produces the stress value s_2.

Third, the change between the current and previous topologies is measured by using their stress values as $(s_2 - s_1)/s_1$. If the change is below a threshold (10%), the candidate point is viewed as a 'good' node, will be added back to the network, and we go back to step 2. Otherwise, at least one of its neighbor connections belongs to a wormhole. We continue to step 4.

Fourth, since all the neighbor connections of the candidate point are independent, we reconstruct the network topology using incremental MDS by adding each neighbor line back and compare the topology change from the previous step. For each line causing the stress value to increase beyond the threshold, a warning packet will be sent to both nodes connected by this line to indicate that this is a false connection and should be removed immediately.

Figure 10 illustrates our detection procedure, which successfully analyzes and identifies all the attacked regions in a distorted network topology. Only several simple user interactions are involved in handling complex wormhole attacks.

Experimental results

The detection accuracy and overhead of the proposed mechanism are evaluated through simulation using net-

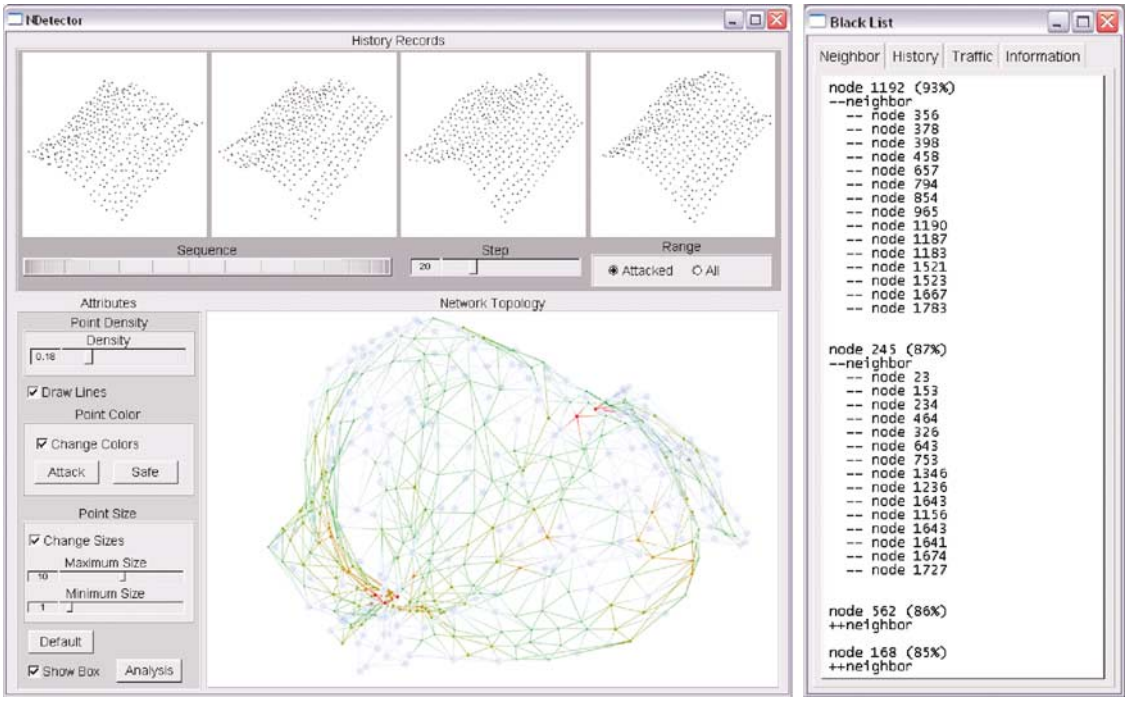

Figure 9 Our wormhole detection interface. The left top locates the history window, left bottom includes parameter window and topology interaction window and the right lists the identities and information of the potential attackers.

work simulator ns_2,[44,45] which is widely adopted by wireless networking investigators. To enable the comparison between IVoW and MDS-VoW and demonstrate the improvements, we adopt a relatively small-scale wireless network. We assume that 600 nodes are randomly and roughly uniformly deployed in a square area with the size of 2 km × 2 km. The communication range among wireless nodes is $r = 180$ m and the average degree of connectivity is 10.35.

We first investigate the relationship between the number of neighbor connection changes and the interval of network reconstructions, so that the tradeoff between the communication and computation overhead of IVoW can be achieved. To enable the comparison between IVoW and MDS-VoW, the same network topology and wormhole attack scenarios are provided to both mechanisms. The detection accuracy is measured by the false alarm rate. Two parameters are of special interest: the fraction of wormholes that are detected, and the number of real neighbor connections that are wrongly labeled as wormholes. Every data point in the following figures represents the average value over 15 trials under different network setups.

Determining the frequency of network reconstructions

The frequency of network reconstructions has a direct impact on the overhead of the proposed mechanism. As we have discussed in the fifth section, incremental MDS uses the previous reconstruction result as the initial layout of the nodes. If too many neighbor relations have changed, more iterations will be required to refine the nodes' positions. On the contrary, every round of reconstruction requires the nodes to measure the distances to their neighbors and send the results to IVoW, which will cause more communication overhead. In this group of experiments, we investigate the relationship between the number of neighbor connection changes and the interval of network reconstructions.

The average lifetime of a neighbor relation is impacted by various factors including the movement model, radio range and moving speed of the nodes. In our simulation, we use the random way point model[46] to describe the movement of the nodes. If the highest moving speed of the nodes is v_{max}, a random value drawn from the uniform distribution $[0, v_{max}]$ will be used as the speed of a node. To compensate the impacts of the differences in moving speeds, we define a unit time = radio range r/v_{max}. The simulation results are illustrated in Figure 11.

We investigate different highest moving speeds ranging from 5 to 20 m/s. From the simulation results, we find that the lifetime of a neighbor connection has a close relationship to the radio range and the moving speed of the nodes. It takes about two units time for every node to change almost all of its neighbors. Therefore, the advantages of incremental MDS can be better demonstrated when the interval of network reconstructions is shorter than two units time.

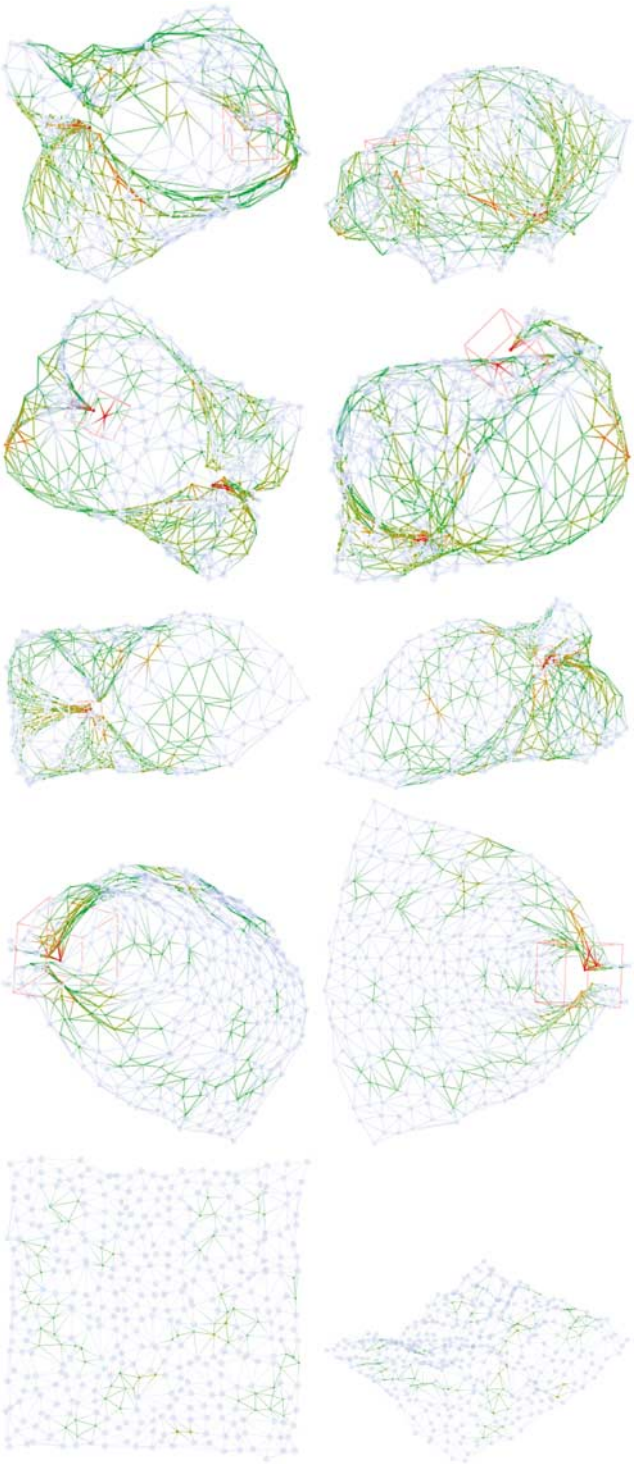

Figure 10 The interactive wormhole detection procedure is shown from top to bottom with two views for each user input and the consequent detection result. A user simply draws a transparent red cube around a potential attack region and a progressive algorithm is performed to analyze the details.

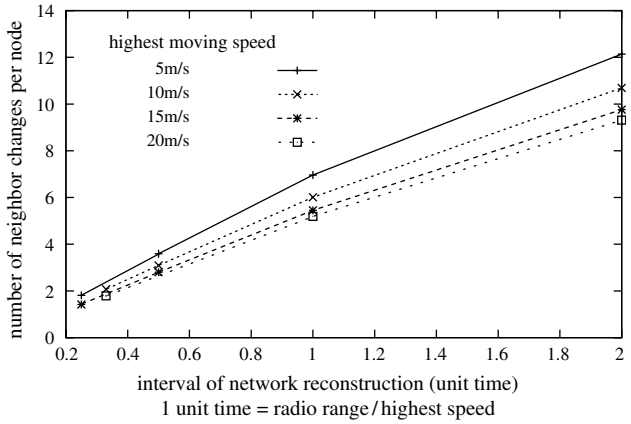

Figure 11 Relationship between neighbor changes and interval of network reconstructions.

Robustness against distance estimation errors

Since network reconstructions are conducted based on the measured distances among wireless nodes, the measurement accuracy has a direct impact on the detection capability. In our simulation, we model the distance estimation errors as uniform noises. If the real distance between two nodes is $d (d \leqslant r)$ and the error rate is e_m, a random value drawn from the uniform distribution $[d \times (1 - e_m), \min(r, d \times (1 + e_m))]$ will be used as the measured distance. We examine different values of e_m from 0 to 80%. We assume that only one wormhole exists in the network and the victims of the attack are randomly selected. The simulation results are illustrated in Figure 12.

The results show that IVoW can greatly improve the detection accuracy when the distance estimation errors are relatively large. The improvements are primarily realized through the user interactions. They take advantage of the expertise and judgments of the user to drastically reduce the size of the suspicious area so that the more complicated detection method described in section Interactive detection can be applied to a localized network.

The false positive alarms will lead to the breaks of real neighbor connections and an increase in the average path length. If all connections of a node are broken, an isolated node will be generated. To examine the impacts of the false positive alarms, we show in Figure 13 the increase in the average path length between all node pairs. Since the degree of connectivity in the original layout is relatively large, the increase in the average path length is small. We do not detect isolated nodes in the experiments.

Detection accuracy under multiple wormholes

One advantage of IVoW is that it can detect fake neighbor connections when there are multiple wormholes in the network. In this group of experiments, we fix the value of e_m at 40% and examine the detection accuracy of the proposed mechanism when the number of wormholes

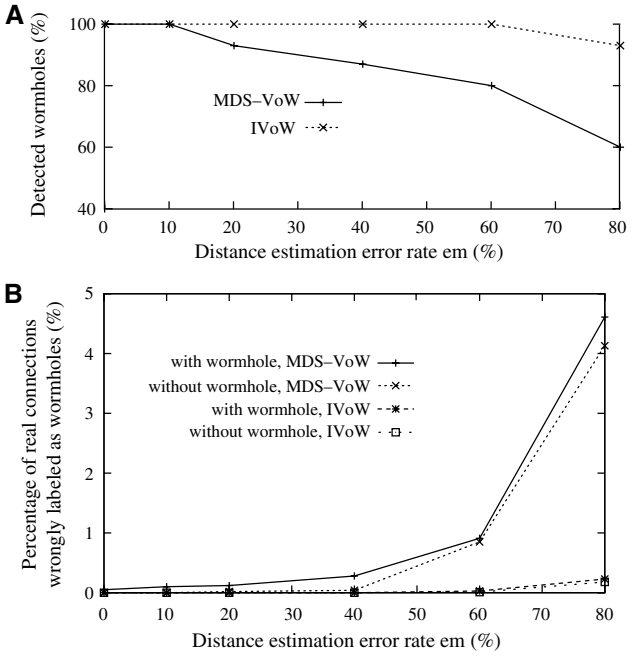

Figure 12 Detection accuracy of IVoW under different distance estimation error rates. (A) Improvements in detection accuracy. (B) Reduction in false positive alarms.

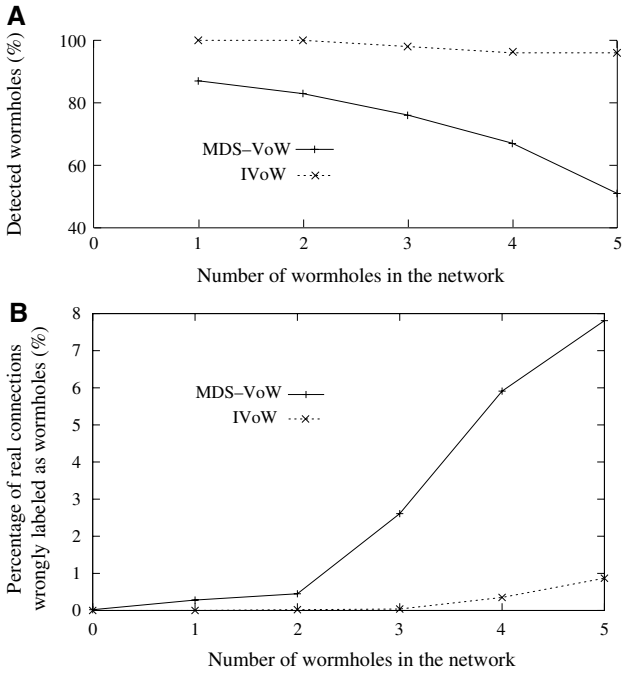

Figure 14 Detection accuracy of IVoW when the number of wormholes increases. (A) Improvements in detection accuracy. (B) Reduction in false positive alarms.

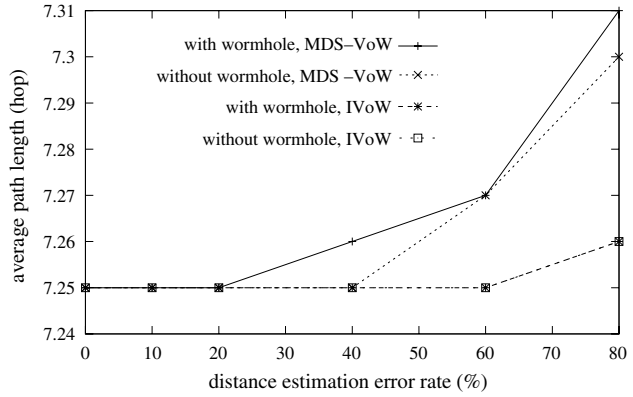

Figure 13 Increase in the average path length caused by false positive alarms.

changes. The victims of the attacks are randomly and independently selected as long as the distance between the two ends of a wormhole is longer than the communication range. Since the increase in the average path length is small in Figure 13, we focus on the false alarm ratio in this group of experiments. The results are illustrated in Figure 14.

The improvements are more obvious in this group of experiments. Through the user interactions, the detection procedure can locate the suspicious areas more accurately and the impacts of multiple wormholes on the detection accuracy are reduced. From the results in Figures 12–14, we find that the user interactions can improve both the wormhole detection efficiency and accuracy, and the proposed mechanism is robust against the distance estimation errors and multiple wormholes.

User study

We have performed initial user studies to evaluate the effectiveness of our visualization approach. We are particularly interested in assessing the acceptance of using visual forms in network intrusion detection tasks. Therefore, we design and analyze the user study focusing on the detection accuracy and duration factors. Below we describe the details of our experiment setup, procedure, data analysis, results, and discussions respectively.

Stimuli

The stimuli of the study are 50 independently generated wireless network scenarios visualized by the proposed approach. We equally divide the 50 trials into 10 groups and introduce different numbers of wormholes into the networks, resulting in five networks in each of the zero-to-nine wormhole scenarios. Each network contains 600 nodes that are randomly and uniformly distributed in a 2 km × 2 km area. The radio range among wireless nodes is 180 m. To simulate real wireless network communications, the distance estimation errors are modeled as

uniform noises and [−40%, 40%] range errors are randomly added to each measured distance.[47] The two end nodes of every wormhole are randomly and independently selected as long as the distance between them is longer than the radio range. The fake distances are also randomly selected from [0, radio range].

Subjects and setup

The subjects of the study include eight volunteers (two females and six males) who are research staff or graduate students in computer science or electrical engineering majors. Half of the subjects have network and security backgrounds while the other half do not. Most of the subjects do not have graphics and visualization backgrounds. All the eight volunteers have normal vision and are not color blind.

The reconstructed network topologies are displayed on a 19-inch LCD monitor with 1280×1024 resolution. The experiment runs in full screen mode with white background and with color, size, and line enhanced visualizations. A general Dell USB 2 button mouse and keyboard are used as the interaction tools.

Procedure

We included a training session before the experiment, since most of our subjects are not familiar with wormhole attacks. During the training session, we briefly introduce the background of this experiment and explain the distortions in the network topology caused by wormhole attacks. Four examples of wormhole attacks, different from all 50 trial data, are selected to arrange the training session parallel to the real experiment. These examples are selected to demonstrate how the attacks will lead to distortions in the reconstructed network and how they are different from the changes caused by distance estimation errors. The subjects are also trained to use our system interface to rotate and move the reconstructed network and select the suspicious areas under attack. There is no time limit on the training session and questions from the subjects are answered.

The experiment is performed right after the training session. Before the experiment, subjects are informed that there are a total of 50 trials. Some are under wormhole attacks and some are normal network scenarios. A subject's task is to find a suspicious region under attack and move the red box provided by the system to enclose it. We fix the size of the red box for all subjects to avoid the effect of a subject choosing a larger size to increase accuracy. A pilot study is performed to determine the suitable cube size, which is large enough to enclose the suspicious regions for all 50 trial data, and as small as possible to measure accuracy of the subjects.

Subjects are also informed that their performances during the experiment will be recorded, although there is no time limit on each trial. The left bottom corner of the system shows the current trial number during the experiment. Since the whole experiment usually takes 20–30 min, we have designed a pause/resume function, which allows the subjects to have a break during the experiment. The time between the pause and resume actions is removed from the recorded duration. The screen turns blank during the break time to prevent the subjects from working on the trial data.

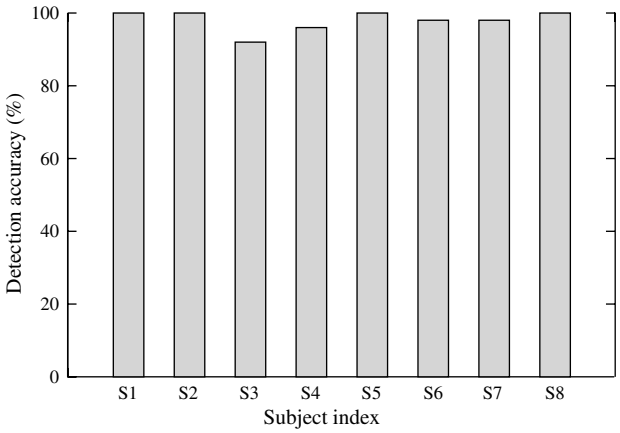

Figure 15 The detection accuracy of each subject over the 50 trials.

During the experiment session, the 50 stimuli network scenarios are presented to the subject in random order. The subject can rotate and move the reconstructed network with the provided interface to get a better understanding of the network structure. Since our proposed detection mechanism is a progressive approach (section Interactive detection), subjects are only asked to identify one suspicious area in each network scenario. Once she/he has located a suspicious area, she/he can move the mouse with the 'm' key pressed to move the fix-sized red cube to enclose the area. When the subject is satisfied with her/his input, she/he can click the 'next' button on the toolbar, which will bring her/him to the next trial. If the subject finds that no wormhole exists in the network, a 'zero' button on the toolbar can be clicked to indicate so, and the red box diminishes from the screen. This process repeats until all 50 trials have been finished.

For each trial, the position of the red box and the indicator of wormhole existence are recorded. The timestamps at the start of the trial, the start of the red box movement (the first time the 'm' key is pressed), and the click of the 'next' button are also captured.

Data analysis

Both detection accuracy and duration are calculated for each trial of every subject. Statistical results are shown in Figures 15–18. Our hypothesis is that subjects will respond reasonably fast and accurately under different wormhole attack scenarios. Therefore, we calculate the average and

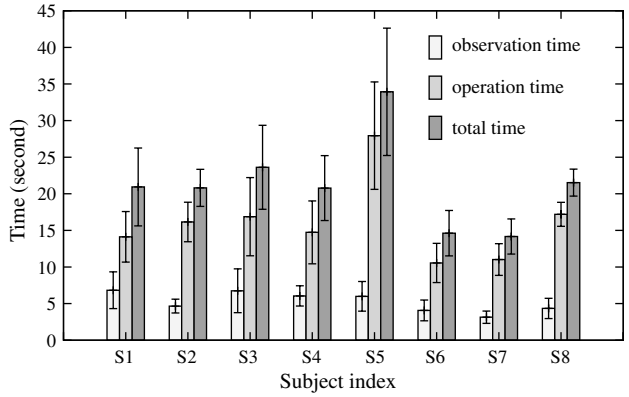

Figure 16 The average observation, operation and total time duration of each subject over the experiment. The operation duration does not include the data from the five trials without wormhole attacks, since they do not require subjects to adjust the red box position. The observation and total time duration are over the 50 trials.

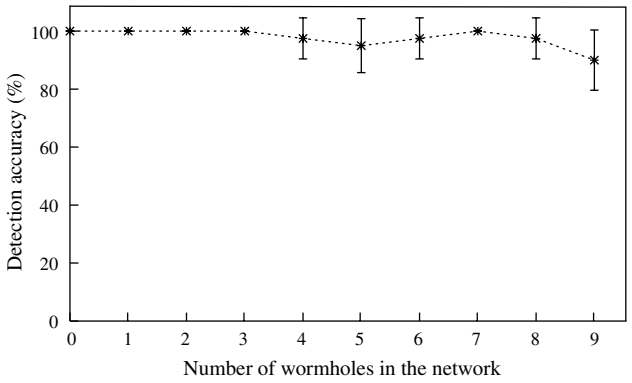

Figure 17 The average detection accuracy under different wormhole attack numbers over the eight subjects.

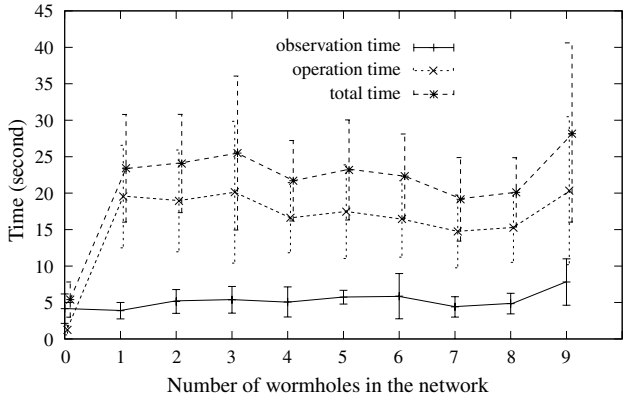

Figure 18 The average observation, operation and total time duration under different wormhole attack numbers over the eight subjects.

standard deviation of the detection accuracy and duration over both the 50 trials and the eight subjects.

The detection accuracy is measured using the recorded indicator of wormhole existence and the position of the red box. For each trial, the wormhole attack information from the corresponding network setup is first compared with the indicator value to see if there is a wormhole in the network. If there exists at least one wormhole, the positions of the nodes linked by wormholes are tested to see if they are inside the red box. Only when both nodes linked by a wormhole are enclosed, will the trial be counted as accurate, otherwise it is considered to be inaccurate.

The total duration of each trial is measured between the beginning of the current and the beginning of the next trial. During the experiment, a subject usually first rotates the network visualization to detect and locate a wormhole. When there are multiple wormholes, some subjects choose an obvious one for simpler operation and some subjects choose the first one they see. Then, they will move the red box to enclose the identified region. Since the network reconstruction is in a 3D space, a subject may rotate the network again to check if both ends of the wormhole are in the red box until she/he is satisfied. Because of this general procedure, we roughly divide the total duration into an observation period and an operation period by the moment the red box is first moved. Since the network scenarios without wormhole attacks do not need the subjects to move the red box, their operation times are not included in the final statistical duration results.

Results and discussions

Since the proposed interactive wormhole detection method incorporates the visual analysis capability of users into the detection procedure, our hypothesis is that our approach would achieve a high detection accuracy. The results in Figure 15 show that all the subjects are above 94% accurate and the average of the eight subjects is 98.25%, which is significantly higher than the previous method.[18] Since the wormhole-free network scenarios do not include any obvious distortions, we assume that these trials are relatively easy to recognize. Figure 17 shows that all the subjects are 100% accurate under this situation. Figure 17 also shows that the average detection accuracy under 1–9 simultaneous wormhole attacks ranges between 92.5 and 100%, with variances of 0 to 0.1. This demonstrates that our detection method can achieve a high detection accuracy under the scenarios of multiple wormhole attacks.

Figure 16 shows that the observation durations of different subjects are similar and the operation durations have a much larger variance. Since the observation durations roughly measure the analysis process of a user, their similarity demonstrates that the features of wormhole attacks illustrated by our visualization method can be identified by all the subjects. The average observation

durations of the eight subjects in Figure 16 are between 3.1 and 6.8 s, and the average observation durations under different numbers of wormholes in Figure 18 are between 3.9 and 7.8 s. This means that all the 8 subjects can provide useful detection suggestions within 8 s even when there are multiple wormhole attacks in the network. This demonstrates that our method can be used to provide accurate detection information rapidly.

The total duration is largely affected by the operation time, since it is 2–3 times the observation duration. The operation duration is mainly decided by whether the subject can fluently interact with the provided interface. For example, subject S5, who has the longest operation duration and still achieves 100% accuracy, claimed that he had very little experience on using a mouse to control 3D scenes. Since we concentrate on the effectiveness of the visualization, the operation time under 30 s is acceptable. We can shorten the interaction time by adopting more intelligent interfaces or new tracking technologies.

Combining the results in Figures 17 and 18, we can see that both the detection accuracy and the observation duration under different numbers of wormhole attacks are not significantly different. This demonstrates that the observation duration to identify a single wormhole and the detection accuracy are robust against the attack complexity (number of wormholes). These results indicate that our method can be used to provide immediate and accurate responses for defending against wormhole attacks.

Conclusions and future work

In this paper, we propose an approach that integrates visual representation, user interaction and automatic analysis algorithms to defend against wormhole attacks in wireless networks. Through integrating interactive visualization into multiple steps of IVoW including representation, monitoring and detection, we show that visualization not only can be used to improve information understanding, but also can be combined with domain knowledge and user expertise to solve problems through visual analysis.

Immediate extensions to the proposed approach include the following aspects. First, to further reduce the overhead and false alarms caused by distance estimation errors, we propose to explore the approaches that can detect attacks on network topology solely based on the connectivity information among wireless nodes. Second, a distributed version of IVoW will be developed to avoid the security problems such as single point of failure. Finally, since the terrain and shape of the area covered by a wireless network can be very complicated, the robustness and error-tolerance of IVoW need to be improved.

While the detection of one kind of attack is investigated in depth, the basic ideas presented in this paper can be extended to deal with other aspects of network security. For example, the anomalies in the localized neighbor relations caused by the fake identities can be used to detect Sybil attacks.[48,49] If the reconstructed network topology is monitored together with the traffic flows, the black holes of data transmission in wireless networks can be located.[50] The visualization techniques and interaction interfaces enable the users to analyze and manage the networks with an ever-increasing scale and complexity. Furthermore, directly applying the network topology information to attack detection avoids the overhead and inaccuracy caused by the parameter measurement procedures.

To better evaluate the proposed approach, we will apply IVoW to real large-scale wireless network environments such as underwater sensor networks, in which the nodes can move freely in 3D spaces. A more generic attack detection framework integrating visualization and interaction techniques will be developed to enforce wireless network security.

Acknowledgements

We thank the anonymous reviewers for their valuable comments. This research is supported by KU New Faculty General Research Fund and Department of Energy under Award Number DE-FG02-06ER25733.

References

1 Zhang Y, Lee W. Intrusion detection in wireless ad-hoc networks. *Proceedings of ACM MobiCom*, Boston, MA, 2000; 275–283.
2 Bhuse V, Gupta A. Anomaly intrusion detection in wireless sensor networks. *Journal of High Speed Networks* 2006; **15**(1):33–51.
3 Du W, Fang L, Ning P. LAD: localization anomaly detection for wireless sensor networks. *Proceedings of International Parallel and Distributed Processing Symposium*, 2005, Denver, Colorado.
4 Fan W, Miller M, Stolfo S, Lee W, Chan P. Using artificial anomalies to detect unknown and known network intrusions. *Knowledge and Information Systems* 2004; **6**(5):507–527.
5 Hall J, Barbeau M, Kranakis E. Using mobility profiles for anomaly based intrusion detection in mobile networks. *Proceedings of Wireless and Mobile Security Workshop* 2005, San Diego, California.
6 Chirumamilla M, Ramamurthy B. Agent based intrusion detection and response system for wireless LANs. *Proceedings of IEEE International Conference on Communications* 2003, Anchorage, AK; 492–496.
7 Zhang Y, Lee W, Huang Y. Intrusion detection techniques for mobile wireless networks. *Wireless Networks Journal* 2003; **9**(5):545–556.
8 Abdullah K, Lee C, Conti G, Copeland J, Stasko J. IDS RainStorm: visualizing IDS alarms. *Proceedings of VizSEC*, 2005, Minneapolis, Minnesota.
9 Fink G, Muessig P, North C. Visual correlation of host processes and traffic. *Proceedings of VizSEC* 2005, Minneapolis, Minnesota.
10 Ren P, Gao Y, Li Z, Chen Y, Watson B. IDGraphs: intrusion detection and analysis using histographs. *Proceedings of VizSEC* 2005, Minneapolis, Minnesota.
11 Goodall J, Rheingans P, Lutters W, Komlodi A. Preserving the big picture: visual network traffic analysis with TNV. *Proceedings of VizSEC* 2005, Minneapolis, Minnesota.
12 Muelder C, Ma K, Bartoletti T. A visualization methodology for characterization of network scans. *Proceedings of VizSEC* 2005, Minneapolis, Minnesota.
13 Dahill B, Levine B, Royer E, Shields C. A secure routing protocol for ad hoc networks. Technical report CS-02-32, University of Massachusetts, 2001.
14 Hu Y, Perrig A, Johnson D. Packet leashes: a defense against wormhole attacks in wireless ad hoc networks. *Proceedings of IEEE NFOCOM* 2003, San Francisco, CA.
15 Papadimitratos P, Haas Z. Secure routing for mobile ad hoc networks. *Proceedings of SCS CNDS* 2002, San Antonio, Texas.

16 Hu L, Evans D. Using directional antennas to prevent wormhole attacks. *Proceedings of NDSS* 2004, San Diego, California.
17 Kong J, Ji Z, Wang W, Gerla M, Bagrodia R, Bhargava B. Low-cost attacks against packet delivery localization and time synchronization services in under-water sensor networks. *Proceedings of ACM Wireless Security (WiSe)* 2005; Cologne, Germany; 87–96.
18 Wang W, Bhargava B. Visualization of wormhole attacks in sensor networks. *Proceedings of ACM Workshop on Wireless Security* 2004, Philadelphia, PA.
19 Basalaj W. Incremental multidimensional scaling method for database visualization. *Proceedings of Visual Data Exploration and Analysis VI, SPIE* 1999; Vol. **3643**:149–158.
20 Williams M, Munzner T. Steerable, progressive multidimensional scaling. *Proceedings of InfoVis* 2004, Austin, Texas.
21 Clark B, Colbourn C, Johnson D. Unit disk graphs. *Discrete Mathematics* 1990; **86**: 165–177.
22 Davison M. *Multidimensional Scaling*. John Wiley and Sons: New York, 1983.
23 Torgeson W. Multidimensional scaling of similarity. *Psychometrika* 1965; **30**: 379–393.
24 Shang Y, Ruml W, Zhang Y, Fromherz M. Localization from mere connectivity. *Proceedings of ACM MobiHoc* 2003, Annapolis, Maryland.
25 Ji X, Zha H. Sensor positioning in wireless ad-hoc sensor networks with multidimensional scaling. *Proceedings of INFOCOM* 2004, Hong Kong.
26 Biswas P, Ye Y. Semidefinite programming for ad hoc wireless sensor network localization. *Proceedings of ACM/IEEE IPSN* 2004, Berkeley, CA.
27 Poovendran R, Lazos L. A graph theoretic framework for preventing the wormhole attack in wireless ad hoc networks. *ACM Journal on Wireless Networks* (WINET) 2007, accepted for publication.
28 Ladd A, Bekris K, Rudys A, Marceau G, Kavraki L, Wallach D. Robotics-based location sensing using wireless ethernet. *Proceedings of ACM MobiCom* 2002, Atlanta, GA.
29 Priyantha N, Chakraborty A, Padmanabhan H. The cricket location support system. *Proceedings of ACM MobiCom* 2000; Boston, MA, 32–43.
30 Savvides A, Han C, Srivastava M. Dynamic fine-grained localization in ad-hoc networks of sensors. *Proceedings of MobiCom* 2001, Long Beach, California.
31 Niculescu D, Nath B. Ad hoc positioning system (APS) using AoA. *Proceedings of IEEE INFOCOM* 2003, San Francisco, CA.
32 Ball R, Fink G, Rathi A, Shah S, North C. Home-centric visualization of network traffic for security administration. *Proceedings of ACM VizSEC* 2004, Fairfax, Virginia.
33 Komlodi A, Rheingans P, Ayachit U, Goodall J, Joshi A. A user-centered look at glyph-based security visualization. *Proceedings of VizSEC* 2005, Minneapolis, Minnesota.
34 Lakkaraju K, Yurcik W, Lee A, Bearavolu R, Li Y, Yin X. NVisionIP: Netflow visualizations of system state for security situational awareness. *Proceedings of ACM VizSEC/DMSEC* 2004, Fairfax, Virginia.
35 Yurcik W. VisFlowConnect-IP: a link-based visualization of Netflows for security monitoring. *18th Annual FIRST Conference on Computer Security Incident Handling* 2006, Baltimore, MD.
36 McPherson J, Ma K, Krystosk P, Bartoletti T, Christensen M. ProtVis: a tool for port-based detection of security events. *Proceedings of ACM VizSEC/DMSEC* 2004, Fairfax, Virginia.
37 Rafiei D, Curial S. Effectively visualizing large networks through sampling. *Proceedings of IEEE Visualization* 2005, Minneapolis, Minnesota.
38 Livnat Y, Agutter J, Moon S, Erbacher R, Foresti S. A visualization paradigm for network intrusion detection. *Proceedings of the IEEE Information Assurance Workshop* 2005; West Point, NY, 92–99.
39 Dijkstra EW. A note on two problems in connexion with graphs. *Numerische Mathematik* 1959; **1**: 269–271.
40 Chalmers M. A linear iteration time layout algorithm for visualizing high dimensional data. *Proceedings IEEE Visualization* 1996; San Francisco, CA; 127–132.
41 Morrison A, Ross G, Chalmers M. A hybrid layout algorithm for subquadratic multidimensional scaling. *Proceedings IEEE Symposium on Information Visualization* 2002; Boston, MA; 152–158.
42 Rao A, Ratnasamy S, Papadimitriou C, Shenker S, Stoica I. Geographic routing without location information. *Proceedings of ACM MobiCom* 2003; San Diego, CA; 96–108.
43 Mucke EP. A robust implementation for three-dimensional Delaunay triangulations. *Proceedings of the First International Computational Geometry Software Workshop* 1995; Minneapolis, MN; 70–73.
44 IEC Workshop on Internet Simulations with the NS simulator 2000, San Diego, CA.
45 The Network Simulator – ns-2 http://www.isi.edu/nsnam/ns/ (accessed 20 July, 2006).
46 Bettstetter C, Resta G, Santi P. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on Mobile Computing* 2003; **2**: 257–269.
47 Savarese C, Rabay J, Langendoen K. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. *USENIX Technical Annual Conference* 2002, Monterey, CA.
48 Douceur J. The sybil attack. *Proceedings of the IPTPS Workshop* 2002, Cambridge, MA.
49 Newsome J, Shi E, Song D, Perrig A. The sybil attack in sensor networks: analysis & defenses. *IPSN'04: Proceedings of the Third International Symposium on Information Processing in Sensor Networks* 2004; 259–268.
50 Marti S, Giuli T, Lai K, Baker M. Mitigating routing misbehavior in mobile ad hoc networks. *Proceedings of Mobile Computing and Networking* 2000; 255–265.