A Vulnerability Study of Geographic Routing in Underwater Acoustic Networks

Michael Zuba, Michael Fagan, Jun-Hong Cui and Zhijie Shi
Department of Computer Science and Engineering,
University of Connecticut, Storrs, Connecticut 06269
{*zuba, fagan, jcui, zshi*}*@engr.uconn.edu*

***Abstract*—Underwater Acoustic Networks (UANs) are utilized in many sensitive commercial, scientific and military applications. However, current network protocols have not been designed to defend against security attacks that can block or degrade network communication and performance. Geographic routing is an essential service used in UANs and current protocols are quite vulnerable. In this paper we call attention to unique factors of UANs that need to be considered when designing secure network protocols and study the vulnerabilities of geographic routing in UANs. We investigate the effects of location spoofing on a standard UAN geographic routing protocol, Depth-Based Routing (DBR), as a case study. We provide a preliminary spoofing attack that can stop network traffic from reaching its destination. Attack performance is given using detailed simulation analysis. Lastly, we improve our attack by using a placement scheme to sample positions in the deployment area in order to increase the attack effectiveness while meeting energy constraints. Our attack is tested on both static and mobile UANs.**

***Index Terms*—Underwater Acoustic Networks, Vulnerability, Security, Geographic Routing Protocols**

I. Introduction

Underwater Acoustic Networks (UANs) have gained considerable attention in recent years. With a majority of our planet covered by water, there is a growing interest in exploring its illusive depths, observing oceanic processes and biology and protecting coastal and port areas. UANs are becoming rapidly accepted as a means to accomplish these tasks as they provide an avenue to introduce new and extend current applications in aqueous environments [1]–[4]. These networks provide a promising solution for efficiently and safely exploring and observing our underwater environments.

UANs make use of underwater acoustic communication channels in order to perform their collaborative tasks and communication efforts. The reason for this form of communication is given by the unique characteristics of the underwater environment which prevents electromagnetic waves from propagating over long distances. However, underwater acoustic communication has many inherent challenges. The propagation speed of acoustic signals in water is about $1.5 \times 10^3 m/s$, which is five orders of magnitude lower than the propagation speed of radio in air ($3 \times 10^8 m/s$). The bandwidth is also limited and dependent upon both operating frequency and transmission range. Additionally, underwater acoustic channels are plagued by path loss, noise generated from passing ships and sea life, multipath and Doppler spread. These factors can create high error probability in the underwater acoustic channel [5].

Significant progress has been achieved on addressing design issues of UANs, such as system integration, communication techniques and networking protocols. However, focus on providing security in these networks has been limited. Malicious adversaries may wish to tamper with these systems by attaching unauthorized nodes or disrupt a subset or potentially all functions of the network. Recent work in [6]–[9] has shown that UANs are vulnerable to denial-of-service (DoS) attacks, with emphasis more on the *physical layer* of the network. DoS attacks can disrupt certain aspects or functions of the network and block communication entirely. This work has spurred an investigation into security concerns for UANs.

In this paper we investigate the effects of malicious adversaries attacking the *network layer*, who want to exploit routing mechanisms. While many routing protocol techniques have been proposed, such as proactive or on-demand routing, our focus is strictly on geographic routing for UANs. Traditional routing protocols, such as AODV, do not work well in UANs because of their costly route discovery process, which is unsuitable given the long-delay characteristics of the underwater acoustic channel [7]. Geographic or geo-routing protocols are preferred because no dedicated route discovery process is needed and packets can be forwarded based on the location information of the network nodes. Additionally, location information is a requirement of most aquatic applications and is a practical parameter (specifically, depth information). However, geo-routing protocols are generally based on the broadcast nature of the acoustic channel. This can create collisions in packet forwarding in which most protocols require self-adaptation techniques to minimize the collision probability if possible. More importantly, from a security standpoint, broadcast based protocols are considered vulnerable to security attacks, given that packet information can be overheard by passive intruders or unauthorized nodes.

In this paper we examine the vulnerabilities of geographic routing in UANs. Our contributions in this paper are as follows:

- Call attention to the unique factors of UANs.
- Investigate the vulnerabilities of existing UAN geographic routing protocols, using Depth-Based Routing as a case study.
- Derive a preliminary attack model specific to geographic

978-1-4799-0895-0/13/$31.00 ©2013 IEEE

routing protocols and present attack performance using simulations.
- Develop an improved attack model to sample locations in the network to maximize attack effectiveness.

The rest of the paper is organized as follows. In Section II we review routing details, present related work and introduce the unique factors of UANs. Section III studies the vulnerabilities of geographic routing protocols and presents our preliminary attack model. Section IV provides our attack performance using simulation analysis and presents an improved version of our attack which is tested on both static and mobile networks. Finally in Section V we provide our conclusions.

II. Background

Routing protocols generally fall into two categories: reactive (or on-demand) routing and proactive routing. However, protocols in these categories do not work well in UANs. In on-demand routing, the routing procedure is initiated by the communication demand at a source node. During route discovery, the source seeks to establish a route towards the destination by flooding route request messages. In UANs this is costly given the long and variable propagation delays (which leads to a higher collision probability), the power hungry operation of acoustic communication and higher bit error rates. This degradation increases as the scale or size of the network increases. Proactive routing, which makes use of routing tables and has nodes constantly updating their tables, also suffers from the same costly issues. Further, the effect of mobility on nodes amplifies the above issues. Many aquatic applications are not constrained to a specific area and deployed nodes often drift due to ocean currents and sea conditions. Network topologies can rapidly change even with small displacements in node positions. This makes multi-hop packet delivery challenging in UANs [2].

In order to provide scalable, efficient and robust routing in UANs, researchers are looking towards geographic routing mechanisms. Geographic routing relies on location information to send data to specific geographic destinations instead of using network addresses [10]. Each node is required to know some location information about itself and the sender has a specific location in mind for the destination of data. Using this technique, a message can be routed to any location without knowledge of the network topology or prior established routes. For the purposes of UANs, geographic routing is becoming the go-to technique in order to solve routing issues. Aquatic monitoring and exploration applications are only useful with location-aware data. This is due to the necessity to associate sampled data with the 3D position it originated from, in order to to spatially reconstruct characteristics of an event [2]. Therefore, the availability of location information is a requirement which helps to enable geographic routing. Further, geographic routing is proven to be more efficient than pure flooding in UANs and helps to lower the impact of node mobility on routing performance [2].

A. Related Work

The security of geographic routing in terrestrial networks has not gained much attention [11] but defenses against potential attackers have been proposed. In [12] the authors survey routing mechanisms for terrestrial networks and discuss potential vulnerabilities. Two geo-routing protocols, Geographic and Energy Aware Routing (GEAR) and Greedy Perimeter Stateless Routing (GPSR) are shown to be insecure against Sybil attacks which forge fake location information. However, because the focus is on general routing mechanisms, no specific analysis of the attacks is given. A resilient geographic routing scheme based on probabilistic multipath routing and trust management is proposed in [11]. Location verification, involving the use of radio signal strength (RSS), time of arrival (TOA), time difference of arrival (TDOA) and angle of arrival (AOA), is used to mitigate false neighbors. Additionally, each node holds a routing table with an associated trust value, which is updated at each transmission. Securing geographic routing in vehicle networks is explored in [13]. Using various sensors in the network, the trustworthiness of a node's claimed position can be estimated. This approach does not need a dedicated infrastructure but does require that every node knows its exact location with use of GPS. Similar work on secure geographic routing in vehicle networks is presented in [14]. This work also addresses secure vehicle communication by using GPS to gather exact position information and then enforces plausibility checks, such as time and velocity requirements, to ensure messages are coming from legitimate locations.

B. Unique Factors of UANs

These existing defenses cannot be directly applied to UANs. This is because of the inherent issues with UAN communication and system constraints [7], [8]. Making use of RSS, TOA, TDOA and AOA in UANs is difficult because of the inaccuracies in estimating these characteristics in acoustic communication. Additionally, GPS does not work underwater, making trust models using exact location constraints difficult to achieve. We can categorize five unique factors of UANs that influence protocol design and are generally not considered in terrestrial sensor networks. These factors are high bit error rates, large and variable propagation delays, narrow bandwidth, computational and energy constraints and a lack of accurate full-dimensional location information. We will also discuss how each factor affects the design of potential security countermeasures.

High bit error rates are the result of underlying channel effects such as multipath and fading. Environmental effects such as water turbulence, currents, ship activity and sea life can also cause high bit error rates. These characteristics affect the link quality between a sender and a receiver which can result in many packets being lost or unable to be decoded at the receiver. In severe scenarios, connectivity between nodes can be completely lost. The effects of high bit error rates can be sporadic or frequent given the dynamic nature of the underwater environment. Channel conditions are rarely the same, even over short time periods. This plays a crucial role in

designing security mechanisms for UANs. A secure protocol scheme should avoid relying on the use of security packets or control/data packets as these packets may get lost. This limits the use of authentication and verification schemes.

Large and variable propagation delays are caused by the speed of sound in water. Acoustic signals in water propagate much slower than radio waves in air. This allows malicious adversaries more time to block or manipulate the communication signal and less time for the network to respond or combat security threats. Further, given a network where nodes are affected by ocean currents, empirical observations propose that underwater sensor nodes move at a speed of 3-6km per hour with an effective dispersivity from 10^{-3} to $10^{3}cm^2/s$ in the vertical direction and from 10^{-3} to $10^{5}cm^2/s$ in the horizontal direction [15]. This node mobility will make propagation delays highly variable impacting potential schemes that might make use of propagation delay knowledge, such as TOA and TDOA.

Additionally, the *narrow bandwidth* of acoustic channels is an issue. This means that the data rate will decrease as the transmission range increases, limiting the amount of information that can be sent or shared and increasing packet size overhead. Secure protocols should further minimize including security information into packets as the usable data rates are already limited.

Another factor is the *power constraints* of UAN systems. While terrestrial sensor networks also have rigid power constraints, UANs are often considered a more extreme case. Acoustic communication is a power hungry operation and consumes more energy to transmit than that of radio communication. Additionally, UANs are deployed in more harsh conditions and most researchers want to maximize deployment lifetime as retrieving these systems is difficult and costly. A security scheme will want to minimize the use of many transmissions and movement (if mobile) in order to secure the network or combat against an attacker.

The final factor is the *lack of accurate full-dimensional location information*. This is especially important to geographic routing and network mechanisms, both security and non-security related. In UANs GPS does not work well underwater and therefore every node (except surface nodes) cannot guarantee reliable positioning information, aside from depth which can be determined easily using a sensor. This impacts schemes that make use of existing security mechanisms which rely on attaining **exact** full-dimensional location information to use verifiers and location checking for trust management schemes. Additionally, networks can be highly dynamic given the potential mobility of network nodes. While localization in UANs is well studied and maturing rapidly, the availability of frequent and accurate full-dimensional positioning information of each node is challenging without assistance from GPS. We emphasize this factor for its impact on defense mechanisms and note that current geographic routing protocols in UANs are quite effective without accurate and full-dimensional location information, such as [16] and [17].

The work in this paper will study the vulnerabilities of geographic routing protocols in UANs. We are not aware of any work to date that involves security aspects in geographic routing protocols for UANs. While traditional security methods, such as cryptography, are an essential network security building block, we note that many security threats, including the one presented in this paper, threaten UANs even with an ideal cryptographic system in place. For the purpose of this work, we assume that no cryptographic system is in place or that it has been breached. Key distribution and authentication schemes come with high communication overhead and latency, are energy and computationally consuming and do not scale well for dense networks. Additionally, even with cryptographic methods, attacks can still be launched against the network. Nodes are generally scattered in a large region that may be unmanned or not monitored by network operators and therefore physical security is jeopardized and nodes can easily be compromised, tampered with and then injected back into the network [7], [8], [18].

III. Vulnerability Analysis

In this paper we will focus on a standard geographic routing protocol for UANs, known as *Depth-Based Routing (DBR)*. DBR [16] is a depth-based protocol that utilizes the unique properties of UANs: specifically, that data sinks are generally located at the water surface. DBR greedily forwards data packets towards the water surface based on the depth information of each node. In DBR, a data packet will record a unique ID when first created and the depth information of its most recent forwarder. The depth information is updated at every hop with the depth of the node forwarding the packet. The basic idea of DBR is that when a node receives a packet for the first time, it will queue this packet for forwarding if that node's depth is smaller than the depth recorded in the packet from the sender (i.e. $d_{receiver} < d_{sender}$) and the depth difference between that node and the sender is greater than some predefined threshold (TH) (i.e. $(d_{sender} - d_{receiver}) >$ TH). Otherwise, it will discard the packet. However, depending on the topology, a packet may be forwarded along multiple paths to the sink or void areas, which is not addressed in DBR.

In order to reduce redundant forwardings, DBR includes a mechanism to suppress redundant packets. This mechanism is known as the *holding time*. When a node receives a packet, there might be multiple nodes in the same area that qualify for forwarding, therefore each node will hold the packet for a certain amount of time after it is queued, the *holding time*. A node that is closer to the surface of the water will have a shorter holding time, resulting in a higher priority to forward the packet. When this node forwards the packet, its neighbors will receive the packet. Any node that receives a duplicate packet during its holding time will check a similar constraint as above. If $d_{receiver} \leq d_{sender}$ and $(d_{sender} - d_{receiver}) >$ TH, it will update its holding time with the minimum holding time of the two. Otherwise, it will drop this packet. Therefore, neighbors at a lower depth (i.e. physically below the optimal receiver), who already have the packet but are in the holding

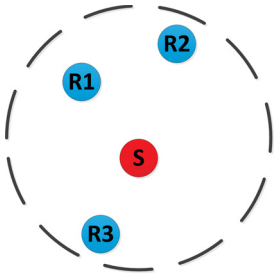

Fig. 1. Visualization of typical DBR scenario

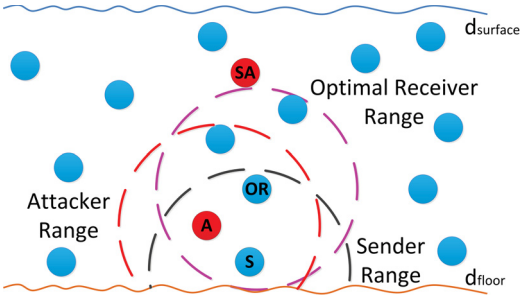

Fig. 2. Visualization of preliminary attack model

phase, will become suppressed and discard the packet. The neighbors at a higher depth will receive the packet and enter their own holding phases. This process will be continued until the packet reaches the intended sink at the surface.

We further explain how DBR works using Figure 1, where we have one sender, *S*, and three receivers, *R1, R2, R3*. The sender will record its depth information into the data packet and broadcast this packet, in which all nodes inside the sender's transmission range (the dashed line) will hear, *R1, R2, R3*. Each receiver will compare their depth with the depth recorded in the packet they receive. In this scenario, *R3* will immediately discard the data packet since it came from a node that is closer to the surface. *R1* and *R2* will compare the depth information and both observe that they are potential candidates to forward the packet. Both nodes will then hold the packet for their calculated holding time. *R2* is closer to the surface than *R1* and therefore has a shorter holding time and will forward the packet first. It is the optimal next hop in the network. Once *R2* forwards the packet, *R1* will receive this packet during its holding time and then discard the original packet from *S* because it is no longer a good candidate to forward.

A. Security Vulnerabilities

Geographic routing protocols route packets based on distance information and not connectivity information. This technique increases security concerns because of the use of location information, which is included in each data packet transmitted in the DBR protocol. Since no governing mechanism exists to verify that a node is in fact at the position it is claiming, malicious attackers can easily exploit the system. The specific weakness, or attacking point, of DBR comes from its heuristics to save energy by deterring redundant transmissions of data packets. The *holding time*, used to schedule the forwarding of data packets, allows for malicious users to exploit this protocol and implement various routing disruptions on the network.

While the work in this paper focuses on the use of DBR as the routing protocol, other geographic routing protocols designed for underwater networks use similar techniques to schedule forwarding. For example, *Vector-Based Forwarding (VBF)* [19] uses a desirableness factor in order to calculate how long the protocol should wait before forwarding a packet, known in VBF as $T_{adaption}$. This is a similar technique compared to DBR's holding time. *HH-VBF* [20] is another protocol using this technique. Further, *HydroCast* [17] uses the broadcast nature of the acoustic channel for nodes to overhear transmissions and calculate their own forwarding priority and then compare it against the distance in that transmission. Therefore, we show that the work in this paper can easily be adapted to other underwater geographic routing protocols and use DBR as a case study.

B. Preliminary Attack Model

In this section we will discuss the attack model for our adversary. We assume that the network of interest is stationary or static. We evaluate the case of mobile nodes in a later section. We also assume that the transmission range for legitimate nodes and the attacker is the same. We formally define $d_{surface}$ as the depth of the surface node with a value of 0 and d_{floor} as the depth of the ocean floor. The network is bounded by d_{floor} and $d_{surface}$ such that all node depths, d, are that of $d_{surface} \leq d_{floor}$. We also define the following, $d_{attacker}$ as the actual depth of the attacker, d_{spoof} as the spoofed or fake depth of the attacker, d_{sender} as the depth of the sender of the message, $d_{optimal}$ as the depth of the receiver or optimal next hop in the network and R as the maximal transmission range of a node. We assume that the attacker is in range of an optimal node and a sender. We start with a network such that $d_{surface} < d_{optimal} < d_{attacker} < d_{sender} < d_{floor}$. This is represented in Figure 2, where S is the sender, A is the attacker's actual position, SA is the spoofed (fake) attacker position and OR is the optimal receiver or next node.

The attacker will start in a passive mode where he will try to eavesdrop network transmissions between at least two nodes in his area. A sender or forwarder sends a message and the optimal receiver will get this message, calculate its holding time and then forward this message at the end of its holding time. Since the optimal receiver always forwards first, all other nodes in the area that received the original message will become suppressed upon hearing the optimal receiver's message. Each packet in DBR will contain the value of $d_{optimal}$ and by having the attacker eavesdrop the transmissions in an area, it can determine the depth of the original sender or forwarder and its optimal receiver (the next forwarder). The attacker can then continue to listen to see if he can hear any more forwarders. In some cases, the

attacker might be able to hear more than one hop given the topology conditions. In either case, the attacker will know the depths of the nodes sending or forwarding. When the attacker receives another packet from the original sender, he can then immediately forward that packet with his fake depth, d_{spoof}, encoded into the packet. Every node in the attacker's transmission range will hear this message and then drop their packets. This is because they received the same packet from a node claiming to be at a better position. It does not matter if the legitimate packet or the attacker's packet is heard first, in either case the packet will be dropped as long as d_{spoof} is a better position than their own. This attack suppresses network traffic and ends the flow of data through this area completely because the attacker has mimicked that he is above them and forwarding upwards.

The choice of d_{spoof} decides the attack performance. Depending on many factors or the amount of knowledge known about the network, attackers may choose different values. If the attacker does not know much about the network at first, to ensure that no node will forward a packet, the attacker should set $d_{spoof} \leq (d_{optimal} - R)$. This will make it such that any node that receives this packet will instantly drop the original packet from its queue or ignore it (if not queued) since the depth in the packet is better than the node that might receive it. This is due to the fact that it fakes the location of a node just outside of the optimal receiver's communication range. Since the attacker itself cannot communicate to the depth of $d_{optimal}$ - R (because $d_{optimal} < d_{attacker}$), it is guaranteed that any node who receives the packet will be at depths greater than the depth in the packet. Therefore, the design of DBR ensures that no one will forward the packet. Further, the constraints mentioned above in Section III-A for a node to be considered a forwarding candidate, enforce a minimum depth bound for the attacker. In order to be successful in attacking the network, the attacker must use a value for d_{spoof} of at least the depth of the optimal receiver (DOR), otherwise it will be ignored. We confirm this and further explore the performance of different values of d_{spoof} in our experiments in Section IV and IV-A.

This attack is powerful because it is easy to perform on a network and also maintains the integrity of the data being transmitted in the network. There is no need to manipulate or modify any data encoded in the packet, other than the depth information, which is already modified at every step of the forwarding process. Essentially, the attacker's packet is the legitimate data but with a fake depth encoded into the packet. This limits the noticeable traces of the attack.

Another item to mention is that the time frame before a node sends or forwards a packet can be large. In order to calculate the holding time (which decides the time frame), DBR uses the following equation:

$$f(d) = 2\tau/\delta \times (R - d) \quad (1)$$

where d is difference between the depth of the sender and the node that received the packet and $\tau = R/v_o$, where v_o is the propagation speed of sound in water, R is the transmission range and δ is a parameter set by the network operator from

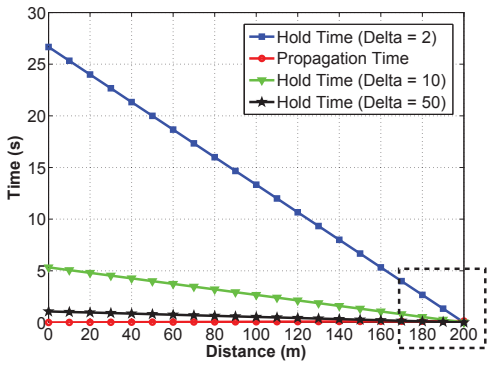

Fig. 3. Comparison of holding times vs. propagation time

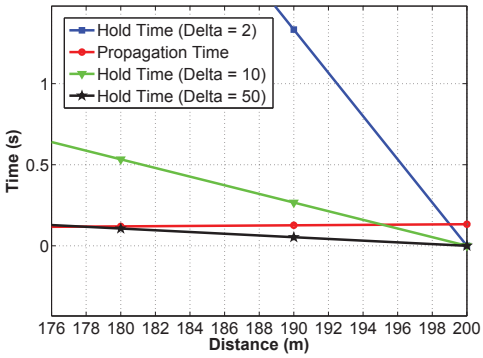

Fig. 4. Intersection of holding times vs. propagation time

1 to R. The δ parameter is used to set how DBR should operate. With a larger value, each node will hold the packet for a shorter time, decreasing average end-to-end delay but increasing redundant forwardings and with a smaller value, each node will have a longer holding time with an increased average end-to-end delay. Obviously, a smaller δ minimizes unnecessary transmissions and decreases energy consumption.

In Figure 3 we compare the holding time using different values of δ (as the difference in depth between the sender and receiver increases) and the propagation time (as distance increases). We note that, given the scale of the graph, it is difficult to observe that the propagation time is in fact increasing over distance. As we observe from this graph, when DBR is not in a flooding mode, i.e. δ is not larger than 50, the holding time is much longer than the propagation time. Therefore, an attacker has a bigger buffer range to adjust its attack or wait longer to be more discrete. While the attacker does not need to know the exact holding time, as we send out our attack packet immediately after receiving a packet, picking a more discrete time to send is an interesting problem of its own. We leave this for future work. However, in Figure 4 we zoom in to the boxed area on Figure 3 and show that when a receiver is nearing the edge of the senders transmission range, its holding time is smaller than the propagation time. Therefore, the attacker has no room for waiting and must send its fake packet immediately.

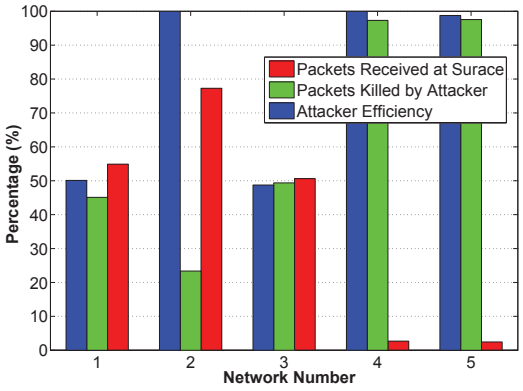

Fig. 5. Simulation results for 5 random networks

IV. Attack Performance

In order to test our proposed attack scheme we have developed a simulation environment with a graphic visualizer that shows the network and how messages are propagating. This environment is used to simulate the DBR routing protocol, the underwater communication channel and the attacker model. We have developed the simulator using Java and due to space limitations we refer the reader to [21] for architecture, implementation and fidelity details.

For our experimental analysis we generated 5 random fully connected networks (meaning that at least one path to the surface exists) in a $1000m \times 1000m \times 1000m$ area. Each deployment had 35 randomly deployed network nodes, 3 surface nodes, 1 randomly deployed attacker close to a network path, a transmission range of $200m$, $\delta = 5$, a send depth = $900m$ and a threshold = $10m$. A send depth means that only nodes deeper than $900m$ will send new data, the rest of the nodes will act as forwarders and a threshold is a parameter set by DBR that sets a constraint on how far away the next node must be in order to forward. In this case, a node must be greater than $10m$ above the sender/forwarder that it received the packet from in order to consider forwarding this packet. The experiment was run for 300 seconds where data packets were sent randomly every 5 to 10 seconds for the first 90 seconds of the experiment, the rest of the time is for packets to finish propagating through the network. We average these results over 50 simulations. The results can be seen in Figure 5, where each trial number is the network number. The blue bar is the attacker efficiency, the number of packets stopped by the attacker divided by the number of packets the attacker received or could stop, the green bar is the number of packets the attacker stopped or killed divided by the total number of packets sent out in the network and the red bar is the number of packets received at the surface nodes divided by the total number of packets sent out. We can draw from these results that the attack works well but that the topology will play an important role in attacker effectiveness.

As mentioned in Section III-B, d_{spoof}, or the faked location, will have an impact on the effectiveness of the attack. We have experimented with various spoofed depths on network 4 from the previous experiment. The settings and simulation details are the same. The results can be seen in Figure 6. In the first experiment, the attacker spoofs a depth of 0. The second experiment has the attacker spoofing the depth of the optimal receiver (DOR) in the routing path minus the transmission range (i.e if the optimal receiver has a depth of $300m$, the attacker spoofs a depth of $100m$). We note that DOR is the depth of the optimal receiver in the attackers range. The third experiment uses a depth of DOR-$10m$, which is a depth just above the optimal receiver (a depth of $290m$ using the previous example). In the fourth experiment, the attacker pretends to be at the same depth as the optimal receiver, the fifth experiment is $1m$ below the optimal receiver and the final experiment is DOR+TH (a depth of $310m$ using the previous examples). The results show that any depth smaller than DOR or DOR itself will perform at almost 100% effectiveness in this network. As expected, depths larger than DOR have no effectiveness. The reason for this is because of the constraints (discussed in Section III) set by DBR on how a node decides if it will forward a packet or not.

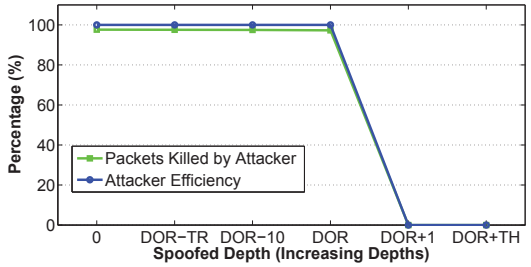

Fig. 6. Simulation results for various spoofed depths

We further tested the use of DOR as the value of d_{spoof} on the other 4 networks from the first experiment. These results were also the same as using a value of DOR-TR for d_{spoof}. These results do not suffer major performance degradation because no other nodes exist in the attackers transmission range that might be a better candidate. In Section IV-A we perform this experiments again using our improved attack model.

Another experiment was performed to analyze the effects of δ in the DBR protocol. This parameter is used to calculate the holding time as shown in Equation 1. Again, a larger value for δ will result in a shorter hold time, increasing redundant forwardings, and a smaller value will result in a longer holding time with reduced forwardings. We tested our attack scheme on network 4, the same network used in the depth experiments. All parameters and simulation settings are the same. We varied δ from 1 to R and each value resulted in the same attack performance. The attackers efficiency was 100% and the number of packets killed was between 96% and 98%. It is clear that δ has no effect on attack performance.

A. Improved Attack Model

Network topologies are highly dynamic. Therefore, we have developed an improved attack model to traverse any given deployment area and find an optimal location to attack

the network. This is a difficult task as the attacker has no knowledge of exactly how well its attack is performing on the network as a whole or even at a given area. However, our approach shows that we do not need to know this information since we can exploit the nature of underwater geographic routing by trying to locate routing paths.

We introduce our approach as follows: the input is a deployment area, which is then partitioned into a grid region by the transmission range of the attacker using a depth of $1m$ as the top plane since we do not want the surface nodes to hear our attacker. We assume the attacker is mobile, in this case an autonomous underwater vehicle (AUV), and can enter the deployment area from any side or the top of the network and can record its movement for future navigation. Once the search area has been input, the attacker will move to the first position in the grid and begin listening for packets to discover routing paths. The goal is elegant: the attacker will traverse across the plane it is on and try to find a single routing path. It will listen for data transmissions in each grid position on the current plane as it traverses across until it reaches the last position on the plane. If the attacker only heard transmissions at one location then it will move to this location, as it is the best spot in the network to attack.

Given the routing nature, from some depth to the surface, discovering the plane with only one routing path implies you have found a bottleneck in the topology as all data must be transmitted through this area. However, if the attacker found more than one routing path, he can set these locations as the new search space bounds and begin searching in the next plane (below or above, dependent on its current location) to try and find a converging path. If the distance between the two furthest boundaries is greater than twice the attackers transmission range then it should go up or down two planes since it is unlikely that those two paths will converge in the next plane given their distance apart. If the attacker moves down and searches between the new bounds and does not find a converging path but finds that he stops hearing traffic at one of the boundaries, he will expand his search in that direction until he picks up the path again, correcting the bounds. If he does not correct this, the attacker could think he found a single path when really one path just moves farther away with depth. In the case that a single path cannot be found, the attacker will move to the location where it heard the most traffic when its energy has reached a threshold used for searching. In cases of multipath routing, when no convergence can be found, the attacker will have to settle for the location with the most observed traffic. This opens up one direction for future work with distributed and collaborative attackers. We make use of existing energy requirements from a well-known AUV, known as the REMUS [22], which can operate for 8 hours at 5 knots or up to 20 hours at 3 knots. We also note that in each experiment, the network is assumed to have been operating before our attacker enters the deployment area.

The results of our improved attack model on the previous 5 network topologies from Section IV can be seen in Figure 7. This figure shows that the attacker was able to find an optimal

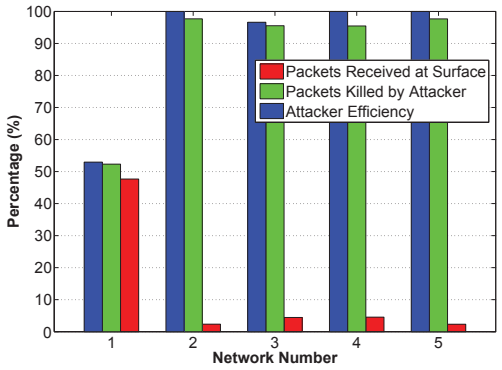

Fig. 7. Improved attack model results for previous 5 networks

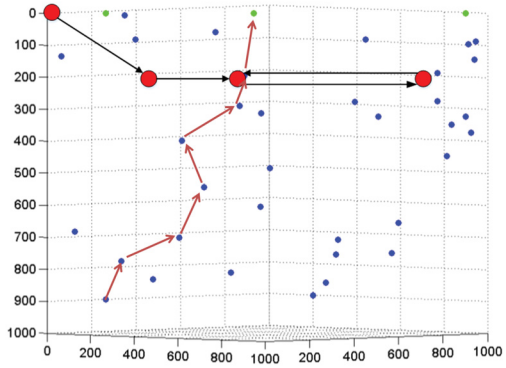

Fig. 8. Search positions and data transmission route

position for all networks, increasing the attack performance over random placement and comes close to a 100% packet kill rate (except in network 1). Networks 2-5 all had convergence points while network 1 had two paths. Therefore, network 1 only allowed for roughly 50% effectiveness. We further summarize our improved attack model performance on these networks in Table I, which shows how long it took (in seconds) for our attacker to find the best position and how many locations were checked before finding the best spot. We note that the time to find a location is a result of how fast the attacker could traverse the network (we use a speed of $2.57222m/s$, taken from the REMUS AUV) combined with the listening time at each position, in these experiments the listening time is 60 seconds. In the case of network 2, the best position was the last position on the plane, while in network 3 the best position was at the first position of the plane and therefore the attacker had to go all the way back to reach this position.

As we observe from the search time, the energy consumption by searching is small (move time in Table I), leaving the rest of the energy life for attacking the network. In Figure 8 we show the positions our attacker searched and the flow of data packets in the network for network 4. We note that the blue nodes are regular network nodes, the green nodes are surface nodes and the red nodes are the attacker positions. We do not show the sender on the right side of the network as it

becomes a void zone (i.e. no path to the surface exists).

Additionally, we wanted to further test how setting d_{spoof} affects the performance of the attack. We ran our improved attack model again on each of the 5 networks and when the best location was found, the attacker started its attack phase. This time, instead of using DOR-TR as the spoofed depth, the attacker used DOR (which is the depth of the legitimate optimal receiver in the area). The results were the same as using a depth of DOR-TR as there were no nodes in the attackers transmission range that had better positions. For static networks, spoofing a depth of DOR is quite powerful, especially given that it mimics the depth of the legitimate receiver.

TABLE I
IMPROVED ATTACK MODEL PERFORMANCE

Network #	Move Time (s)	Total Time (s)	Positions Checked
1	388.769	588.769	4
2	233.26	383.26	3
3	544.27	724.27	3
4	466.52	646.52	3
5	311.01	491.01	4

B. Impact of Mobility

Until now, we have analyzed our attack performance on static networks. In this section we will analyze our improved attack on a mobile network using a kinematic model from [23] known as the Tidal Mobility Model, which captures the chaotic stirring in tidal areas. We approximate this mobility similar to [24] and is as follows:

$$\begin{cases} V_x = k_1 \lambda \upsilon \sin(k_2 x)\cos(k_3 y) + k_1 \lambda \cos(2k_1 t) + k_4 \\ V_y = -k_1 \lambda \upsilon \cos(k_2 x)\sin(k_3 y) \end{cases} \quad (2)$$

where V_x is the speed in the x axis and V_y is the speed in the y axis. Additionally, k_1, k_2, k_3, λ, and υ are variables closely related to environmental factors such as tides and bathymetry. These values change in different environments. As well, k_4 and k_5 are random variables. We assume k_1 and k_2 are random variables which are subject to normal distribution with π as the mean and $(0.1\pi)^2$ as the standard deviation; k_3 is subject to the normal distribution with 2π as the mean and $(0.2\pi)^2$ as the standard deviation; λ is subject to the normal distribution with 1 as the mean and 0.01^2 as the standard deviation; υ is subject to the normal distribution with 0.2 as the mean and $(0.01)^2$ as the standard deviation; and k_4 and k_5 are subject to the normal distribution with 0.1 as the mean and $(0.01)^2$ as the standard deviation. The effect of mobility in the 2D case is shown in Figure 9 where the red node is the starting location and the green node is the end location after $6000s$. This model was originally produced for 2D networks and therefore we adopt a 3D version of this model.

We use the same network, network 4, as used in our previous experiments. The simulation settings and parameters are the same except for the value of δ, which has increased to shorten holding times for mobile networks. We assume the senders are anchored to the bottom of the sea floor, otherwise, network connectivity is largely affected. For clarity,

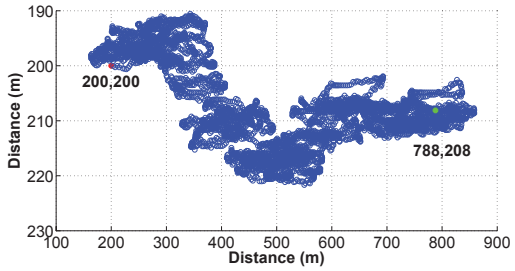

Fig. 9. One example of node mobility over 6000 seconds

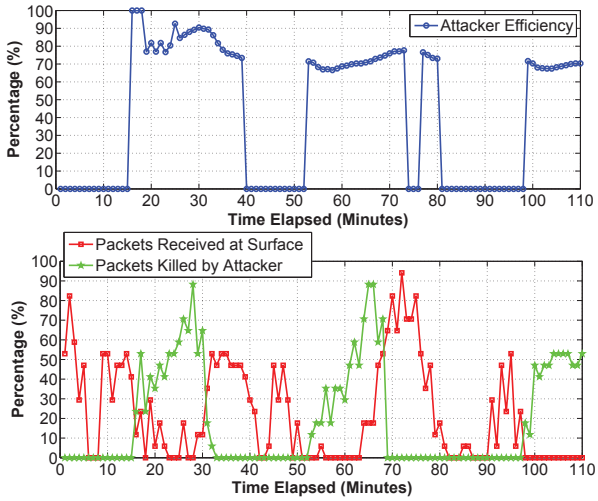

Fig. 10. Performance of improved attack on a mobile network

the settings are a $1000m \times 1000m \times 1000m$ area deployment area with 35 randomly deployed network nodes, 3 surface nodes, a transmission range of $200m$, $\delta = 25$, a send depth = $900m$ and a threshold = $10m$. Given the mobility of the nodes, if a node reaches the surface it is randomly redeployed back into the network and if a node hits a side boundary it is placed on the opposite side boundary at the same depth location. This ensures 35 network nodes in the deployment area and helps to maintain network connectivity. We assume the network has been operating for some time before the attacker node enters the area. Once the attacker finds its optimal location, it will then launch its attack.

We gather attack performance over the course of 110 minutes. The results can be seen in Figure 10. The top figure shows the attacker efficiency over time, where anything lower than 100% means the attacker is not stopping all the messages it could have stopped, except when the efficiency is at 0%, which means that the attacker is not hearing any messages in his area. The bottom figure shows the number of packets that were killed by the attacker and the number of packets received at the surface. We note that network connectivity is lost between the time period of minutes 5-8, 42-44, 48-49, 51-52, 82-84 and 87-91. As the figure shows, mobility impacts the attacker performance. Looking at the attacker efficiency, when it begins to slope down, such as between the time period of minutes 33-40, it shows that nodes are slowly moving out

of the attackers transmission range and the attack is slowly worsening, until all forwarding nodes move out of the area and no more nodes in his area are forwarding data or exist. The same applies for nodes moving into the attackers area, such as between the time period of minutes 50-75 when the attack is getting better and better each minute. In some cases network connectivity is slightly affected, such as between the time period of minutes 100-110. The attacker is killing roughly 50% of the packets being sent but no packets are being received at the surface, implying that the other 50% are lost due to connectivity issues along the way.

TABLE II
IMPROVED ATTACK MODEL PERFORMANCE ON A MOBILE NETWORK

Move Time (s)	Total Time (s)	Positions Checked
4629.996	4959.996	6

We can observe from this figure based on the time periods that there appears to be two general routing paths that the nodes move between. The search time can be seen in Table II. The search time is large because the attacker tried to find a convergent path but was affected by the mobility of nodes. Additionally, the position that was determined to be the best was a much earlier sampled location and therefore roughly 35% of the total time was the attacker moving back to that position. It is clear that for mobile networks, in order to obtain more effective performance the attack model needs to be tuned accordingly. One potential avenue is mobility prediction similar to what is used in localization techniques. We leave this notion for future work.

V. CONCLUSION

In this paper we have called attention to unique factors of UANs and studied the vulnerabilities of geographic routing protocols in UANs. We have proposed a preliminary spoofing based attack model for underwater geographic routing protocols, using DBR as a case study. We provided detailed simulation analysis on attack performance using various network topologies and studied the performance of different spoofed depths. Additionally, we introduced an improved attack model for static networks that locates the best position in a given topology to launch an attack while maintaining energy requirements. Our attack is shown to be powerful and minimal in terms of traces left behind, because no data needs to be manipulated or compromised. Our improved attack was further tested against mobile networks and performs well but could be improved by considering movement predictions.

ACKNOWLEDGMENT

This work is supported by the U.S. National Science Foundation (NSF) under Grant No. 1228936.

REFERENCES

[1] J. Partan, J. Kurose, and B. N. Levine, "A Survey of Practical Issues in Underwater Networks," in *Proc. of the 1st ACM International Workshop on Underwater Networks (WUWNet)*, 2006, pp. 11–24.

[2] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "Challenges: Building Scalable Mobile Underwater Wireless Sensor Networks for Aquatic Applications," *IEEE Network, Special Issue on Wireless Sensor Networking*, vol. 20, no. 3, pp. 12–18, 2006.

[3] J. Kong, J.-H. Cui, D. Wu, and M. Gerla, "Building Underwater Ad-hoc Networks and Sensor Networks for Large Scale Real-time Aquatic Application," in *Proc. of IEEE Military Communications Conference (MILCOM)*, 2005, pp. 1535–1541.

[4] Z. Jiang, "Underwater Acoustic Networks - Issues and Solutions," *International Journal of Intelligent Control and Systems*, vol. 13, no. 3, September 2008.

[5] L. Liu, S. Zhou, and J.-H. Cui, "Prospects and Problems of Wireless Communication for Underwater Sensor Networks," *Wiley Wireless Communications and Mobile Computing, Special Issue on Underwater Sensor Networks*, vol. 8, no. 8, pp. 977–994, 2008.

[6] M. Zuba, Z. Shi, Z. Peng, J. Cui, and S. Zhou, "Vulnerabilities of Underwater Acoustic Networks to Denial-of-Service Jamming Attacks," in *Wiley Security and Communication Networks*, 2012, pp. 1–10.

[7] J. Kong, Z. Ji, W. Wang, M. Gerla, R. Bagrodia, and B. Bhargava, "Low-cost Attacks against Packet Delivery, Localization and Synchronization Services in Under-Water Sensor Networks," in *ACM Workshop on Wireless Security (WiSe)*, 2005, pp. 87 – 96.

[8] R. Zhang and Y. Zhang, "Wormhole-Resilient Secure Neighbor Discovery in Underwater Acoustic Networks," in *Proc. of the 29th IEEE International Conference on Computer Communications (INFOCOM)*, 2010.

[9] M. Zuba, Z. Shi, Z. Peng, and J.-H. Cui, "Launching Denial-of-Service Jamming Attacks in Underwater Sensor Networks," in *Proc. of the 6th ACM International Workshop on Underwater Networks (WUWNet)*, 2011.

[10] R. Flury and R. Wattenhofer, "Randomized 3D Geographic Routing," in *Proc. of the 27th IEEE International Conference on Computer Communications (INFOCOM)*, 2008.

[11] K. Liu, N. Abu-Ghazaleh, and K. Kang, "Location Verification and Trust Management for Resilient Geographic Routing," *Journal of Parallel and Distributed Computing*, vol. 67, no. 2, pp. 215–228, 2007.

[12] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Elsevier Ad Hoc Networks*, vol. 1, no. 2-3, 2003.

[13] T. Leinmuller, C. Maihofer, E. Schoch, and F. Kargl, "Improved Security in Geographic Ad Hoc Routing through Autonomous Position Verification," in *Proc. of the 3rd ACM International Workshop on Vechicular Ad Hoc Networks (VANET)*, 2006.

[14] C. Harsch, A. Festag, and P. Papadimitratos, "Secure Position-Based Routing for VANETs," in *Proc. of the 66th IEEE Vehicular Technology Conference (VTC)*, 2007.

[15] W. Broecker and T.-H. Peng, "Tracers in the Sea." in *Eldigio Press, Lamont Doherty Earth Observatory of Columbia University*, 1982.

[16] H. Yan, Z. Shi, and J.-H. Cui, "DBR: Depth-Based Routing for Underwater Sensor Networks," in *Proc. of IFIP Networking*, 2008.

[17] U. Lee, P. Wang, Y. Noh, L. F. M. Vierira, M. Gerla, and J.-H. Cui, "Pressure Routing for Underwater Sensor Networks," in *Proc. of the 29th IEEE International Conference on Computer Communications (INFOCOM)*, 2010.

[18] Y. Noh, U. Lee, P. Wang, B. Choi, and M. Gerla, "VAPR: Void Aware Pressure Routing for Underwater Sensor Networks," in *IEEE Trans. on Mobile Computing (TMC)*, 2012.

[19] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks," in *Proc. of IFIP Networking*, 2006.

[20] N. Nicolaou, A. See, P. Xie, J.-H. Cui, and D. Maggiorini, "Improving the Robustness of Location-Based Routing for Underwater Sensor Networks." in *Proc. of OCEANS 2007*, 2007.

[21] M. Zuba and M. Fagan, "GeoSim User Manual," in *UConn CSE TR13-04*, 2013.

[22] WHOI, "Autonomous Underwater Vehicle, REMUS," in *[Online: http://www.whoi.edu/osl/remus-auv]*, 2012.

[23] S. Beerens, H. Ridderinkhof, and J. Zimmerman, "An Analytical Study of Chaotic Stirring in Tidal Areas," *Elsevier Chaos, Solitons and Fractals*, vol. 4, no. 6, pp. 1011–1029, 1994.

[24] Z. Zhou, Z. Peng, J. Cui, Z. Shi, and A. Bagtzoglou, "Scalable Localization with Mobility Prediction for Underwater Sensor Networks," *IEEE Trans. on Mobile Computing*, vol. 10, no. 3, pp. 335–348, 2011.