# R-CARP: A Reputation Based Channel Aware Routing Protocol for Underwater Acoustic Sensor Networks

Angelo Capossele
Dipartimento di Informatica
Sapienza Università di Roma
Rome, Italy
WSENSE s.r.l., Rome, Italy
capossele@di.uniroma1.it

Gianluca De Cicco
Dipartimento di Informatica
Sapienza Università di Roma
Rome, Italy
decicco@di.uniroma1.it

Chiara Petrioli
Dipartimento di Informatica
Sapienza Università di Roma
Rome, Italy
WSENSE s.r.l., Rome, Italy
petrioli@di.uniroma1.it

## ABSTRACT

In this paper we introduce R-CARP, a reputation based channel aware routing protocol for underwater acoustic sensor networks (UASNs). While many routing protocols have been proposed for UASNs, solutions to secure routing protocols from attacks such as sinkhole attack and selective forwarding are still overlooked. These routing attacks can dramatically disrupt network performance, especially in some application scenarios such as homeland security and critical infrastructure monitoring, where a high reliability on message delivery is required. Designing secure and reliable protocols for UASNs is particularly challenging due to acoustic modems unique characteristics such as low bandwidth and bit rate, high propagation delays and high energy consumption when in transmit mode. The aim of this work is therefore to propose R-CARP, a secure and reliable routing protocol tailored to such communication constrained environment. R-CARP is an improved version of CARP, the channel aware routing protocol presented in [5], enriched with a reputation based mechanism to contrast malicious node behavior. To secure R-CARP we employ BLS, a short digital signature algorithm, exploiting its aggregation property to reduce the additional communication overhead. By means of simulation based performance evaluation, we show that, under attack, R-CARP is effective at bypassing malicious nodes and outperforms CARP in terms of packet delivery ratio (PDR) and energy per bit (EPB) by a factor of up to 2, at the cost of a slight increment in terms of latency.

## Categories and Subject Descriptors

D.4.6 [**Security and Protection**]: Cryptographic controls; C.2.2 [**Network Protocols**]: Routing protocols.

## General Terms

Security, Routing


*WUWNET* '15, October 22-24 2015, Washington DC, USA.


## Keywords

Underwater security, underwater sensor networks, underwater protocols, reputation based routing, digital signatures, sinkhole attack.

## 1. INTRODUCTION

Underwater Acoustic Sensor Networks (UASNs) have become an important area of research which has driven the proliferation of many applications, including coastal protection, monitoring and discovery of the marine environment, and prediction of underwater seismic and volcanic events [15]. Due to the strong attenuation of radio communication underwater and to the limited range and operation environments of optical communication, acoustic is the typical communication technology of choice. Recently, several solutions have been proposed in the literature at all layers of the protocols stack [5, 21, 19, 23, 1, 4, 6] to enable the realization of these applications. However, the broadcasting nature of the wireless acoustic channel makes the data vulnerable to being eavesdropped, modified or dropped by an attacker, so that the design of secure communication protocols is of paramount importance.

Recent works [3, 11] present a complete suite of cryptographic techniques tailored to the unique features of UASNs that make routing protocols secure against external attacks such as Sybil attack, hello flooding, acknowledgment spoofing, exhaustion and reply attack. Although the solutions proposed in such works provide confidentiality, authentication and integrity of messages, they are vulnerable to insider attacks such as the wormhole and the sinkhole attacks.

Authors in [27, 26] focus on the wormhole attack, by which an attacker uses secret wormhole links to make distant nodes believe to be neighbors. They propose to exploit the Direction of Arrival (DoA) estimation to protect neighbor discovery protocols from the wormhole attack. Multi-path routing and Software Defined Network techniques are presented as security defenses against jamming attack and node failures in [14] and [24] respectively. However, none of the above works have focused on the sinkhole attack [13], where malicious nodes send fake routing information to their neighbors in order to attract the largest possible number of messages. This attack often precedes the selective forwarding or the blackhole attack. In the former the compromised node selectively decides which message to drop, while in the latter every message received by the compromised node is dropped.

The sinkhole attack has been extensively studied in terres-

trial networks, such as Wireless Sensor Networks (WSNs), Mobile Ad-hoc Networks (MANETs) and Delay Tolerant Networks (DTN). To solve the sinkhole attack, authors in [17] and [16] present solutions for WSNs and MANETs based on intrusion detection and cooperative techniques respectively, while authors in [12] propose a reputation based mechanism.

However, the unique features of UASNs do not allow to adopt traditional solutions employed in terrestrial networks [8, 7, 2, 10, 22], due to the high communication overhead needed to detect misbehavior of compromised nodes. In this paper we overcome these limits by proposing R-CARP, a secure and reliable routing protocol tailored to UASN communication-constrained environment. Our protocol is an improved version of CARP, the channel aware routing protocol presented in [5], enriched with a reputation based mechanism to contrast malicious node behavior, and with cryptographic techniques such as the BLS short digital signature algorithm [9] to ensure authentication and integrity, producing a low communication overhead. Our solution is also robust and able to deal with several malicious nodes that collude to perform the sinkhole attack. To the best of our knowledge, R-CARP is the first routing protocol for UASNs which is immune to the sinkhole attack.

The rest of the paper is organized as follows. Section 2 details R-CARP. In Section 3 we present results of a simulation based performance evaluation of our solution. Finally, Section 4 concludes the paper.

## 2. R-CARP

Our proposed protocol R-CARP is built upon CARP [5], a cross-layer routing protocol that exploits link quality information for data forwarding. Although a recent version of CARP proposed in [3] provides support for security properties such as confidentiality, authentication, integrity and protection against external attacks, CARP is still vulnerable to insider attacks such as the sinkhole attack. As shown in Figure 1, if one node is malicious or it has been compromised, it can increase the probability to be chosen as a relay by advertising to its neighbors a high value of the utility function, which is based on hop distance from the sink, available buffer space, residual energy and quality of the link. This type of insider attack can be disruptive when performed in conjunction with the selective forwarding attack, where the compromised node acts as relay for several nodes and decides selectively which packet to drop.

We solve these security issues by proposing R-CARP, a reputation-based routing protocol which analyzes the behavior of nodes and improves the overall network security, by rejecting routing paths containing selfish or malicious nodes that do not cooperate in routing and by choosing the best relay among several neighbors based on the reputation of the potential relay.

### 2.1 Security primitives

To guarantee authentication and integrity of the protocol, R-CARP employs digital signatures. In particular, we have selected the Boneh-Lynn-Shacham [9] (BLS) scheme, which produces short signatures of 160 bit providing a security level of $2^{80}$. BLS is a Pairing Based Cryptography [18] algorithm which defines $E/\mathbb{F}_q$ as an elliptic curve over a finite field $\mathbb{F}_q$, $E(\mathbb{F}_q)$ as the set of points of this curve, and $\#E(\mathbb{F}_q)$ as the order of the group. Let $n$ be a positive integer, $\mathbb{G}$ an additively-written group of order $n$ with identity

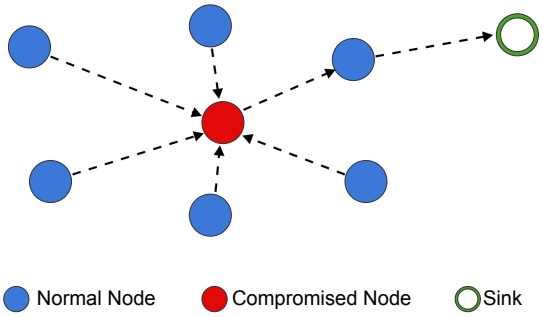



Figure 1: Sinkhole attack.

$\infty$, and $\mathbb{G}_T$ a multiplicatively-written group of order $n$ with identity 1. A bilinear pairing $e$ is an efficiently computable and non-degenerative mapping $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ such that $\forall P, Q \in \mathbb{G}$ and $\forall a, b \in \mathbb{Z}^*$,

$$e([a]P, [b]Q) = e(P, [b]Q)^a = e([a]P, Q)^b = e(P, Q)^{ab}.$$

In addition, the BLS signature scheme defines a map-to-point hash function $H_{map} : \{0,1\}^* \rightarrow \mathbb{G}$ that maps a message to a point of the elliptic curve. Given a secret key $x \in \mathbb{Z}_q^*$, a public key $P_{pub} = xg$ such that $g \in \mathbb{G}$ is the generator of the group $\mathbb{G}$ and a message $m \in \{0,1\}^*$, a signature $\sigma$ can be computed as $\sigma = xH_{map}(m)$ and verified if $e(g, \sigma) = e(P_{pub}, H_{map}(m))$. The BLS scheme provides the signatures aggregation feature, thus, signatures from different signers and on distinct messages can be accumulated into a single signature. For example, given two signers $x$ and $y$, two messages $m_x$ and $m_y$ can be cumulatively signed by means of a point addition operation performed as $\sigma_c = \sigma_x(m_x) + \sigma_y(m_y)$. The signature $\sigma_c$ can be verified if $e(g, \sigma_c) = e(P_{pub_x} + P_{pub_y}, H_{map}(m_x) + H_{map}(m_y))$.

This feature is of paramount importance in UASNs, where communication is characterized by a very high energy consumption of acoustic transmissions and limited bandwidth, thus, it is crucial to send short messages.

### 2.2 The R-CARP protocol

R-CARP protocol starts with a set-up phase that allows the network nodes to acquire hop distance information from the sink. Each node shares the same group key and a unique secret key with the sink. When a node $x$ has data packets to forward, it broadcasts a request message (`PING`) to choose the best suitable relay among its neighbors. Nodes receiving the transmitted request reply with a `PONG` response message containing estimated information on hop distance from the sink, and quality of the link. Link quality information are estimated according to recent history collected at each node about the number of control and data packets correctly received.

For each `PONG` message that node $x$ receives, it calculates the reputation $R_{xy}$ for a node $y$ as:

$$R_{xy} = \frac{packetConfirmed_{xy}}{packetSent_{xy}},$$

where $packetConfirmed_{xy}$ is the number of messages confirmed by the sink that node $x$ has forwarded through node $y$ according to recent history, and $packetSent_{xy}$ is the total number of messages that node $x$ has forwarded through

node $y$ according to recent history.[1] The higher $R_{xy}$, the higher is the reputation of node $y$ according to the local information of node $x$. $R$ values can range between 0 and 1. Node $x$ selects the most suitable relay $y$ among its neighbors by computing an utility function $U_{xy}$ defined as:

$$U_{xy} = \alpha HC_y + \beta Lq_{xy} + \gamma R_{xy},$$

where $HC_y$ is the estimated hop distance of node $y$ from the sink and $Lq_{xy}$ is the estimated quality of the link computed as the sum of link quality between $x$ and $y$ and of the best link quality among those from $y$ to its neighbors $z$. Both $HC_y$ and $Lq_{xy}$ range between 0 and 1. High values of $HC_y$ correspond to a lower hop count distance from the sink, and high values of $Lq_{xy}$ correspond to a better link quality. Note that `PONG` messages are encrypted and authenticated as in [3], thus an attacker cannot alter information on hop distance, link quality, of other nodes but simply lie on its own (this latter case is handled by R-CARP by identifying and bypassing misbehaving nodes).

Once the node $x$ has selected the best relay $y$, it sends the data packet $pkt_i$ to $y$ along with the information of the chosen path $p(x, y)$ (e.g., the sequence of the traversed nodes IDs), authenticated by means of the digital signatures algorithm BLS. More specifically, node $x$ digitally signs the data packet $pkt_i$ as:

$$\sigma_{pkt_i} = \sigma_x(pkt_i, p(x, y))$$

In fact, if node $x$ does not authenticate the data packet along with the chosen path, a compromised node $y$ can modify the data packet or the path, thus, breaking the integrity of messages or altering the update process. In the latter case, a node $y$ can pretend to have been selected as relay (when it was not) or not to have been selected (when it was).

Node $x$ caches in a list $L_{pkt_x}$ both the data packet $pkt_i$ and its path until the delivery is confirmed by the sink.

When the node $y$ successfully receives the data packet from $x$, it replies with an `ACK` packet, selects its best relay $z$, forwards the data packet $pkt_i$ along with the updated information of the chosen path (e.g., $p(x, y, z)$), updating the signature as:

$$\sigma_{pkt_i} = \sigma_x(pkt_i, p(x, y)) + \sigma_y(pkt_i, p(y, z))$$

It is important to highlight that by exploiting the aggregation property of BLS, the signature $\sigma_{pkt_i}$ has always the size of a single signature (20B), thus introducing a fixed communication overhead which does not depend on the number of hops.[2]

Once the data packet is received by the sink, the sink verifies the authenticity of its traversed path and consequently signs a confirmation message $C_{pkt_i}$ for that packet which is sent piggybacked in the `ACK` packet. In order to minimize overhead, reputation information updating is performed reactively by propagating back confirmation messages piggybacked in `PONG` messages. More specifically, when a node $x$ has to forward a new data packet $pkt_j$, it sends along with the `PING` message a list with the packet IDs of its interest (e.g., data packets cached in the list $L_{pkt_x}$). Once node $x$ receives `PONG` messages, it locally increases the reputation of its neighbors based on confirmation messages received piggybacked in `PONG` messages and removes from the list $L_{pkt_x}$ data packets already confirmed. Reputation information of a node $y$ is locally decreased by a node $x$ if a confirmation message for a data packet $pkt_i$ with traversed path $p(x, y)$ is not received before a period of $T_x$ seconds.[3] In this case, it puts $pkt_i$ back in the forwarding queue.

We highlight that even if a node $x$ decreases the reputation of a node $y$ when the latter is not a compromised or malicious node (e.g., data packet has been lost due to high network traffic load or low overall link quality conditions of the traversed path), the degraded performance may indicate that a malicious node could be present in the traversed path. The reputation mechanism of R-CARP guarantees that after few `PING`/`PONG` exchanges, network nodes are able to detect if a sinkhole attack is running, either on data messages or confirmation messages, and start to ignore and isolate the compromised node/path.

# 3. PERFORMANCE EVALUATION

In this section, we evaluate the performance of R-CARP while on both normal and attack conditions, by means of a simulation-based evaluation. R-CARP is the first routing protocol for UASNs immune to the sinkhole attack. Thus, we assess the impact on performance of its overhead when compared to CARP and we evaluate the ability of R-CARP to limit performance degradation in presence of insider attacks, through the analysis of the following metrics.

- *End-to-end latency*, defined as the time between the packet generation and the time of its correct delivery at the sink.
- *Packet delivery ratio*, defined as the ratio of the number of packets correctly delivered to the sink over the total number of packets generated by the nodes.
- *Energy per bit*, defined as the energy consumed by the network to correctly deliver a bit of data to the sink[4].

We also display the ratio of dropped packets in scenarios where malicious nodes perform a blackhole attack, to show that R-CARP is able to effectively bypass such nodes.

Both R-CARP and CARP protocols have been implemented in SUNSET [20] on top of ns-2, connected to the Urick acoustic channel model [25].

In the following we first describe the selected scenarios and protocol parameters settings (Section 3.1). We then report on the results of our simulation experiments according to the different scenarios (Section 3.2).

## 3.1 Simulation scenarios and settings

We consider a static UASN with 9 nodes (8 nodes plus the sink which is located at a side of the deployment area, on the surface). Underwater nodes are randomly and uniformly

[1] In order to avoid that an attacker can exploit a compromised node with a high reputation, R-CARP computes the reputation based on recent nodes behavior, thus, an eventual attack can be quickly detected.

[2] The only overhead information which depends on the number of hops is the traversed path, which can be reduced up to $\log(N_{max})$ byte per hop, where $N_{max}$ is the maximum number of nodes in the network.

[3] This period depends on the estimated RTT of a data packet for each node.

[4] This metric does not take into account the energy spent by the sink, which is typically more powerful than the other nodes.

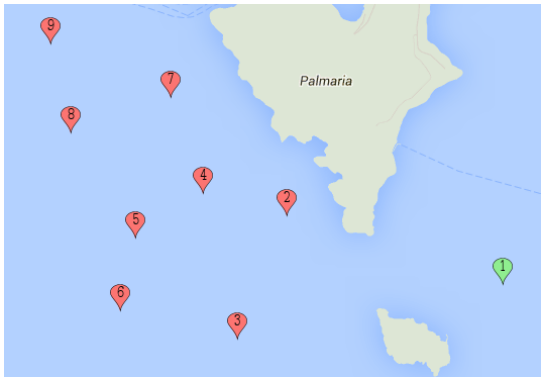



Figure 2: Network topology.

placed in a region with surface of 2 km × 1 km at 50m of depth.

We simulate three different scenarios where network condition depends on the attacker behavior.

- *Normal conditions*, where none of the network nodes is compromised by an attacker.
- *Sinkhole attack*, where, for each simulation, an attacker randomly picks one node to compromise among the set of nodes at 1 hop of distance from the sink.[5]
- *Sinkhole attack with colluding nodes*, where an attacker randomly compromises up to 50% of the network nodes, selecting them among the set of nodes at 1 hop of distance from the sink.

Nodes generate traffic according to a Poisson process with a fixed sample rate of $\lambda$ packets per second, where $\lambda$ takes values in the set $\{0.002, 0.004, 0.008, 0.016, 0.032\}$. The destination of all packets is the sink. Packets are forwarded through the network with an average number of hops from source nodes to the sink of 2.1 and with a maximum number of hops of 4 (Figure 2).

Compromised nodes perform the sinkhole attack by advertising to their neighbors the best value of hop distance from the sink and quality of the link. Once they are selected as relays, they perform the blackhole attack by dropping the packet.

In order to simulate different application scenarios we consider data packets with a payload size of 100 and 250 bytes. The physical header overhead changes according to the data rate but it is dominated by a 10ms synchronization preamble. The medium access control headers of both R-CARP and CARP are 4B long.

In R-CARP data packets require an additional overhead of up to 24B for the digital signature (20B) used to authenticate data and to store compressed information to identify the traversed path (4B). Both protocols share the same size of `PING`, `PONG` and `ACK` packets (11B, 7B and 6B respectively). In addition, R-CARP confirmation messages size is at most 24B (20B of digital signature plus 4B of the traversed path), piggybacked in the `PONG` or `ACK` packets.

We assume a BPSK modulation. The carrier frequency is 25.6kHz for a bandwidth of 5000Hz and a data rate of 5000b/s. We estimate reception and transmission power consumption based on the energy consumption of existing acoustic modems. The transmission power consumption for short control packets (`HELLO`, `PING`, `PONG` and `ACK`) and data packets is set to 3.3W and 8W, respectively. The reception power consumption is set to 0.5W. Finally, we assume that the sink knows the public keys of all the network nodes to verify all the digitally-signed messages, while each node knows the public key of the sink to verify the confirmation messages signed by the sink.

[5]In fact, in order to attract the largest number of packet, the sinkhole attack is typically performed near the sink.

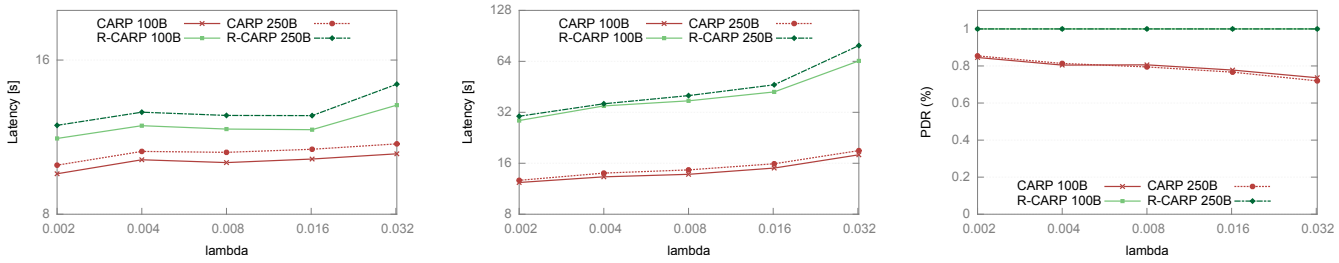
## 3.2 Simulation results

In this section, we compare performance of R-CARP with respect to CARP during: 1)normal conditions; 2)sinkhole attack; 3)sinkhole attack performed by up to 50% of colluding nodes. Figures 3a and 3b show the result of the end-to-end latency during normal and sinkhole attack conditions respectively. Considering the case of normal conditions (Figure 3a), R-CARP presents latency ranging from 11.2s (11.9s) to 13s (14.3s) for a data traffic generation rate of $\lambda = 0.002$ and $\lambda = 0.032$ respectively with data packets size of 100B (250B). R-CARP results are similar to CARP results, where the data latency ranges from 9.6s (10s) to 10.3s (11s). As expected, R-CARP latency increases when a sinkhole attack is running, as shown by Figure 3b. More specifically, latency ranges in R-CARP from 28.6s (30.3s) to 42.3s (46.5s), when the packet size is 100B (250B) while in CARP it ranges from 12.3s (12.7s) to 18s (19s). The higher latency of R-CARP with respect to CARP is due to several reasons. First, the communication overhead introduced by the digitally signed traversed path included in each data packet and the confirmation messages. Second, the presence of a compromised node near the sink decreases the number of good relays available while increases the network traffic load due to the higher number of packets that a good relay has to handle. Third, CARP reduced latency in case of an attack reflects degraded PDR performance which decreases network load.

CARP,during normal conditions presents a packet delivery ratio (PDR) at the sink always greater than 95% during normal conditions, as shown in Figure 3c. Its PDR, however, drops from 84% (85%) down to 71% (73%) during a sinkhole attack, when considering data packets of 100B (250B). This is because CARP nodes trust on the utility function advertised by the compromised node. As a consequence, the compromised node will be selected as relay with high probability, and it will then drop all received data packets. On the other hand, R-CARP presents a PDR of 100% in all considered scenarios.

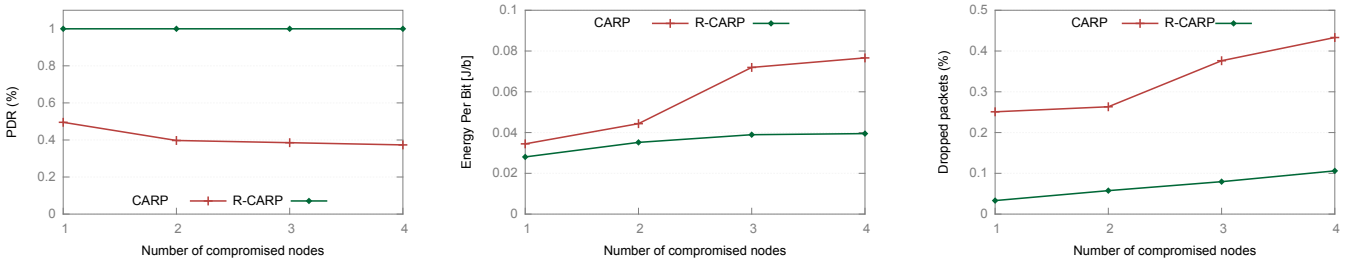
The last set of Figures show the comparison between R-CARP and CARP during a sinkhole attack performed by up to $n$ colluding compromised nodes, with $n$ ranging from 1 to 4 (50% of network nodes, excluding the sink, compromised). Traffic generation is set to $\lambda = 0.016$ and the data payload size to 100B. Figure 4a shows the packet delivery ratio at the sink. R-CARP presents a PDR of 100%, while CARP drops its PDR from 48.5% when the nearest node to the sink is compromised, to 39.3% when 4 nodes are compromised. As expected, R-CARP is able to detect a sinkhole attack and isolate compromised nodes, assuring a perfect PDR even when 50% of network nodes are compromised.

As a consequence, Figure 4b shows higher values of energy per bit for CARP with respect to R-CARP. More specifi-




(a) Latency during normal condition. (b) Latency during sinkhole attack. (c) PDR during sinkhole attack.

Figure 3: R-CARP vs CARP: end-to-end latency during normal and sinkhole attack conditions with different data payload size and variable traffic (3a, 3b). Packet delivery ratio with different data payload size and variable traffic during a sinkhole attack (3c).




(a) PDR with $n$ compromised nodes. (b) Energy per bit with $n$ compromised nodes. (c) Ratio of dropped packets with $n$ compromised nodes.

Figure 4: R-CARP vs CARP: packet delivery ratio (4a), energy per bit (4b) and ratio of data packets dropped (4c), with data payload size of 100B and $\lambda = 0.016$, during the sinkhole attack performed by up to $n$ colluding nodes.

cally, CARP has an energy per bit of 0.035J, 0.043J, 0.073J, 0.078J while R-CARP has an energy per bit of 0.027J, 0.035J, 0.038J and 0.038J when 1, 2, 3 and 4 compromised nodes are colluding. In Figure 4c we show the ratio of data packets that reach a compromised node and are therefore dropped. Due to the reputation mechanism implemented by R-CARP, the percentage of dropped packets ranges from 3% to 10% while in CARP it varies between 25% and 43%. This shows that R-CARP is able to well identify and bypass malicious nodes. End to end reliability is in any case enforced by the confirmation scheme, as demonstrated by the 100% PDR performance.

# 4. CONCLUSIONS

The unique characteristics of UASNs, such as high energy consumption in transmission, low data rate and high computation capability, require different security solutions for the sinkhole attack with respect to terrestrial networks such as WSNs and MANETs. In this work we propose R-CARP, a secure and reliable routing protocol tailored to such communication constrained environment. R-CARP is an improved version of CARP, designed to contrast insider attacks by using a reputation based mechanism and BLS, a short digital signature algorithm for data and traversed path authentication. R-CARP exploits BLS aggregation property to reduce the additional communication overhead. Results of a simulation based performance evaluation show that, under attack, R-CARP outperforms CARP in terms of packet delivery ratio (PDR) and energy per bit (EPB) by a factor of two, at the cost of a slight increment in terms of latency.

# 5. ACKNOWLEDGMENTS

This work has been partially supported by the EC FP7 FIRE IP project SUNRISE “Sensing, monitoring and actuating on the UNderwater world through a federated Research InfraStructure Extending the Future Internet” and by the PRIN project TENACE.

# 6. REFERENCES

[1] J. Alves, R. Petroccia, and J. R. Potter. MPR: Multi-Point Relay Protocol for Underwater Acoustic Networks. In *Proceedings of ACM WUWNet ’14*, pages 16:1–16:8, New York, NY, USA, 2014.

[2] G. Ateniese, G. Bianchi, A. T. Capossele, and C. Petrioli. Low-cost Standard Signatures in Wireless Sensor Networks: A Case for Reviving Pre-computation Techniques? In *Proceedings of NDSS 2013*, San Diego, CA, USA, Febraury 24-27 2013.

[3] G. Ateniese, A. T. Capossele, P. Gjanci, C. Petrioli, and D. Spaccini. SecFUN: Security Framework for Underwater acoustic sensor Networks. In *Proceedings of MTS/IEEE OCEANS 2015*, pages 1–9, Genova, Italy, May 2015.

[4] S. Azad, P. Casari, and M. Zorzi. Multipath routing with limited cross-path interference in underwater networks. *IEEE Wireless Communications Letters*, 3(5):465–468, Oct 2014.

[5] S. Basagni, C. Petrioli, R. Petroccia, and D. Spaccini. CARP: A channel-aware routing protocol for underwater acoustic wireless networks. *Ad Hoc Networks, Special Issue on Advances in Underwater Communications and Networks*, Available on-line, August 2014.

[6] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic. Optimized packet size selection in underwater wireless sensor network communications. *IEEE Journal of Oceanic Engineering*, 37(3):321–337, July 2012.

[7] G. Bianchi, A. T. Capossele, A. Mei, and C. Petrioli. Flexible Key Exchange Negotiation for Wireless Sensor Networks. In *Proceedings of ACM WiNTECH 2010*, Chicago, Illinois, USA, 2010.

[8] G. Bianchi, A. T. Capossele, C. Petrioli, and D. Spenza. AGREE: exploiting energy harvesting to support data-centric access control in WSNs. *Ad hoc networks*, 11(8):2625–2636, 2013.

[9] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. *Journal of Cryptology*, 17(4):297–319, 2004.

[10] A. T. Capossele, V. Cervo, G. De Cicco, and C. Petrioli. Security as a CoAP resource: an optimized DTLS implementation for the IoT. In *Proceedings of IEEE ICC 2015*, London, UK, June 2015.

[11] G. Dini and A. Lo Duca. A secure communication suite for underwater acoustic sensor networks. *Sensors*, 12(11):15133–15158, 2012.

[12] G. Dini and A. Lo Duca. Towards a reputation-based routing protocol to contrast blackholes in a delay tolerant network. *Ad Hoc Networks*, 10(7):1167–1178, Sept. 2012.

[13] M. C. Domingo. Securing underwater wireless communication networks. *IEEE Wireless Communications*, 18(1):22–28, 2011.

[14] M. Goetz, S. Azad, P. Casari, I. Nissen, and M. Zorzi. Jamming-resistant Multi-path Routing for Reliable Intruder Detection in Underwater Networks. In *Proceedings of ACM WUWNet '11*, pages 10:1–10:5, New York, NY, USA, 2011.

[15] J. Heidemann, M. Stojanovic, and M. Zorzi. Underwater sensor networks: applications, advances and challenges. *Philosophical Transactions of the Royal Society A*, 370(1958):158–175, 2012.

[16] J. Jebadurai, A. Melvin, and I. Jebadurai. Sinkhole detection in mobile ad-hoc networks using mutual understanding among nodes. In *Proceedings of ICECT 2011*, volume 3, pages 321–324, April 2011.

[17] I. Krontiris, T. Dimitriou, T. Giannetsos, and M. Mpasoukos. Intrusion detection of sinkhole attacks in wireless sensor networks. In *Algorithmic Aspects of Wireless Sensor Networks*, volume 4837, pages 150–161. Springer Berlin Heidelberg, 2008.

[18] A. Menezes. An introduction to pairing-based cryptography, 2005.

[19] L. Pescosolido, C. Petrioli, and L. Picari. A multi-band noise-aware mac protocol for underwater acoustic sensor networks. In *Proceedings of IEEE WiMob 2013*, Lyon, France, October 2013.

[20] C. Petrioli, R. Petroccia, J. R. Potter, and D. Spaccini. The SUNSET framework for simulation, emulation and at-sea testing of underwater wireless sensor networks. *Ad Hoc Networks, Special Issue on Advances in Underwater Communications and Networks*, Available on-line, August 2014.

[21] C. Petrioli, R. Petroccia, and M. Stojanovic. A Comparative Performance Evaluation of MAC Protocols for Underwater Sensor Networks. In *Proceedings of MTS/IEEE OCEANS 2008*, Quebec City, Canada, September 15-18 2008.

[22] S. Sciancalepore, A. Capossele, G. Piro, G. Boggia, and G. Bianchi. Key Management Protocol with Implicit Certificates for IoT Systems. In *Proceedings of ACM MobiSys 2015 workshop: IoT-Sys 2015*, Florence, Italy, May 18 2015.

[23] C. Tapparello, P. Casari, G. Toso, I. Calabrese, R. Otnes, P. van Walree, M. Goetz, I. Nissen, and M. Zorzi. Performance Evaluation of Forwarding Protocols for the RACUN Network. In *Proceedings of ACM WUWNet '13*, pages 36:1–36:8, New York, NY, USA, 2013.

[24] G. Toso, D. Munaretto, M. Conti, and M. Zorzi. Attack Resilient Underwater Networks Through Software Defined Networking. In *Proceedings of ACM WUWNet '14*, pages 44:1–44:2, New York, NY, USA, 2014.

[25] R. Urick. *Principles of Underwater Sound*. McGraw-Hill, 1983.

[26] W. Wang, J. Kong, B. Bhargava, and M. Gerla. Visualisation of wormholes in underwater sensor networks: a distributed approach. *International Journal of Security and Networks*, 3(1):10–23, 2008.

[27] R. Zhang and Y. Zhang. Wormhole-resilient secure neighbor discovery in underwater acoustic networks. In *Proceedings of IEEE INFOCOM 2010*, pages 1–9, March 2010.