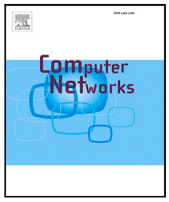
Computer Networks 243 (2024) 110311

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



A RNN-based approach to physical layer authentication in underwater acoustic networks with mobile devices☆

Francesco Ardizzon [a,*], Paolo Casari [b,c], Stefano Tomasin [a,c]

[a] DEI, University of Padova, Via Gradenigo 6/b, 35131, Padova, Italy
[b] DISI, University of Trento, Via Sommarive 9, 38123, Trento, Italy
[c] CNIT, Viale G.P. Usberti, 181/A, 43124, Parma, Italy

ARTICLE INFO

Keywords:
Underwater acoustic networks
Physical layer authentication
Sensor networks

ABSTRACT

Underwater acoustic communications are becoming a popular solution for underwater data communications and telemetry, making the authentication of transmitted data a necessity. In this paper, we propose a physical-layer authentication strategy for underwater acoustic networks (UWANs) with mobile devices. Such a scenario is more challenging than classical authentication scenarios in static networks, because the mobility of the receiver and/or transmitter implies that channel conditions slowly change over time. Thus, we cannot rely on the statistics of channel features to be stationary. In our proposed strategy, we assume that the receiver can rely on a set of sensors. We first extract a set of channel features, to be used to track the channel evolution over time. We then develop a long short-term memory (LSTM)-based approach, where at each step the sensors predict future feature values based on a learned model and on previously observed feature values. Next, each sensor computes the prediction error and passes it on to the actual receiver, which makes a decision on the signal authenticity through a generalized likelihood ratio test (GLRT). We model different classes of attacks and test them using simulation data obtained via the Bellhop ray tracing software. Numerical results show that our authentication mechanism successfully distinguishes between legitimate and impersonating transmitters, even when considering challenging attacking scenarios where the attacker can successfully mimic the channels between the legitimate transmitter and the sensors.

1. Introduction

Underwater acoustic networks (UWANs) exploit acoustic waves to enable communications among devices under the sea surface. Nevertheless, the challenging underwater environment often constrains the attainable data rate, and complex signal processing techniques are needed at the receiver to handle the typically long channel impulse responses, Doppler spread, and interference from other underwater sources [1]. In this context, security protocols functioning at higher protocol stack layers may yield additional overhead (e.g., adding signatures on the message) that may reduce the already limited data rate. Consequently, there has been a significant focus on physical layer security (PLS), which operates at the physical layer and relies on the statistical properties of the channel to provide security.

We tackle the authentication problem, where a receiver acts as a verifier and must decide whether a received message has been transmitted by the legitimate transmitter or from an impersonating attacker. PLS authentication algorithms can be divided into two classes: source-based, and channel-based. In the source-based class, the receiver uses unique features added by the transmitter, (e.g., by its hardware) as a *fingerprint* to authenticate the transmission. Instead, with channel-based authentication, the receiver uses the channel impulse (or frequency) response estimated from the received signal as a fingerprint. This fingerprint is typically unique to the transmitter's and receiver's location [2]. Hence, the channel impulse response measured by an attacker not co-located with the legitimate transmitter will look significantly different from the expected one. Thus, an initial received packet, whose authenticity has been established by higher-layer cryptographic techniques, provides the reference channel response, while the subsequent packets, that are not protected by higher-layer authentication mechanisms, are accepted as authentic if the newly estimated channel response matches the reference one.

☆ This work received support from the NATO Science for Peace and Security Programme under grant no. G5884 (SAFE-UComm). This work was also partially supported by the European Union under the Italian National Recovery and Resilience Plan (PNRR) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE0000001 - program "RESTART").

* Corresponding author.
E-mail addresses: francesco.ardizzon@phd.unipd.it (F. Ardizzon), paolo.casari@unitn.it (P. Casari), stefano.tomasin@unipd.it (S. Tomasin).

https://doi.org/10.1016/j.comnet.2024.110311
Received 6 December 2023; Received in revised form 7 February 2024; Accepted 7 March 2024
Available online 11 March 2024

However, in UWANs, devices are subject to continuous movements induced, among others, by either the natural drift due to sea waves and currents or the transmitter/receiver movements themselves, e.g., when the transmitter and/or the receiver are autonomous underwater vehicles (AUVs). Under these circumstances, the variation of the channel over time can be significant, and any assumption on the channel's time stability becomes unrealistic.

In this paper, we propose a PLS authentication technique for underwater acoustic communications to be used by an UWAN, specifically designed to account for the time variability of the channel. To this end, we consider a set of features extracted from the power delay profile as authentication parameters.

These features include:

- the number of channel taps;
- the average tap power;
- the root mean square (RMS) delay;
- the smoothed received power;
- the power-weighted average tap delay.

This set of features was chosen since they are strongly related to the positions of both the transmitter and the receiver [3,4]. The features can be used to effectively distinguish between legitimate transmitter and attacker by detecting anomalies on the measured channel, e.g., by observing a channel that is associated with a position too far from the position estimated in the previous step. Thus, they constitute a good starting point for physical layer authentication (PLA). We remark that the considered features belong to a channel-based authentication context. This enables us to avoid, for instance, training the authentication algorithm on a specific transmitter device (as would be instead required when working with source-based features). Yet, the proposed approach does not rely on the specific characteristics of the features, hence it may be possible to add features from a source model context as well.

Instead of evaluating the features' statistics, we evaluate their evolution over time, by computing a measure of coherence between currently measured features, and previously measured ones. To do so, we apply a recurrent neural network (RNN), specifically a long short-term memory (LSTM) model, to track the changes in the authentication features. Each sensor will then measure the prediction error and share it with the receiver. The authentication check is then performed by comparing a combination of the errors with a given threshold. Indeed, large discrepancies between the predicted and the observed features indicate an irregular behavior that hints at the start of a possible attack.

The main contributions of this paper are listed as follows.

- We propose a novel channel feature tracking strategy based on RNNs, that makes it possible to track channel features without explicitly requiring knowledge about the transmitter's movements.
- We merge the local prediction error estimates to verify the authenticity of the transmission, based on the generalized likelihood ratio test (GLRT).
- We test the performance of the proposed architecture via Bellhop simulations, considering three attack classes: I) a static attacker; II) a dynamic attacker whose trajectory follows the same model as the legitimate receiver; and III) an attacker trying to replicate the legitimate channel.

In more detail, in attacks I and II the attacker transmits signals from its position, i.e., as if pretending that the legitimate transmitter has moved to its location. Conversely, with attack III, the attacker forges fake signals in an attempt to impersonate the legitimate transmitter. It does so by exploiting a (partial) knowledge of the legitimate channel, and by preprocessing its own transmission to make them as similar as possible to the legitimate ones.

Our authentication approach provides a fundamental security primitive for mission-critical scenarios. Among other examples, such scenarios include: tactical settings; applications where coordination among different actors may cause damage if disrupted; as well as trustful data collection for environmental or equipment/infrastructure monitoring. For instance, consider coordinated underwater vehicles: authenticated message exchanges may avoid that bogus commands by an attacker lead vehicles progressively off course, or into a collision. In the case of data collection, e.g., from critical underwater infrastructure, unwanted abnormal readings or alarms sent by an attacker can be rejected before they lead to costly or disproportionate reactions. Moreover, physical layer techniques help achieve secure authentication in multi-tenant contexts, where heterogeneous underwater assets interact and cannot be assumed to integrate the same hardware cipher.

This work extends the strategy we presented in [4]. In our previous work, we considered a Kalman filter-based approach, where the innovation of the filter provides an authentication feature. However, such an approach is limited, as it can be used only for features that are a function of the distance between the transmitter and receiver. Additionally, it requires a-priori knowledge about the transmitter (and receiver) trajectory. In this paper, instead, we propose a method that does not require knowledge about the relative movement between transmitter and receiver, and that exploits a much broader set of features, thus being more robust and secure. Even if an attacker knows the actual distance between the two devices, it remains hard to predict, e.g., the number of channel taps without an accurate knowledge of the surrounding environment.

The rest of the paper is organized as follows. Section 2 surveys related work on the authentication of underwater transmissions, including physical layer approaches. Section 3 describes the system model. Section 4 details our proposed authentication algorithm. Section 5 presents the simulation results. Finally, Section 6 draws concluding remarks.

2. Related work

Several methods have been investigated to achieve authentication in UWANs [5,6]. The straightforward approach would be to use the cryptography algorithms used for terrestrial cabled or wireless networks. However, Souza et al. explored the communication and computation energy toll that terrestrial network authentication primitives may take if directly applied to underwater network nodes for end-to-end authentication [7]. The authors concluded that short and aggregate signature schemes are recommended in underwater networks.

The work in [8] proposes a secure protocol suite for UWANs. As a part of this suite, the authors advocate the use of message authentication codes [9] to preserve message integrity, at the expense of increasing the message length.

In [10], a game-theoretic approach fosters cooperation among network nodes, by motivating them to improve the effectiveness of end-to-end authentication schemes, which are seen as a key functionality of future UWANs [11].

To reduce the complexity of underwater authentication, Yuan et al. employ matrices of known structure as part of the process, to reduce their memory occupancy and the computational cost of the authentication algorithm [12]. The proposed scheme achieves up to four orders of magnitude less complexity than the standard RSA-based authentication. With a similar purpose, Al Guqhaiman et al. propose a multi-factor scheme based on zero-knowledge proofs via message authentication codes [13]. Specifically, the codes depend not only on pre-shared information but also on communication-related features such as the MAC address of the node, the direction of arrival information, as well as the hop count of the sender. Receiving a packet for which this data does not match any of the features of the receiver's neighbors causes the receiver to label the packet as malicious, and to send an alert to its own network neighborhood.

Zhang et al.'s approach [14] revolves around classical authentication schemes based on message exchanges and mandates the use of lightweight primitives such as chaotic maps and hash functions.

While being slightly lighter than competing schemes from the literature from a computational point of view, the proposed scheme requires less storage to work. Along the same line, in [15] the attacker impersonates multiple network nodes at once (also known as a Sybil attack). Here, the legitimate nodes attempt to identify the attacker's malicious behavior via its node ID and the data stored in the cluster head, which feeds a hierarchical fuzzy system-based trust management model.

Recently, physical layer security approaches have been considered both for authentication and for other security primitives such as key exchange. Physical layer authentication often relies on the collection of channel characteristics (e.g., features of the channel impulse response) to tell apart transmissions by legitimate network members from transmissions by an impersonating attacker.

Considering an underwater LOS environment with negligible multipath, Khalid et al. propose that the receiver keep a database of angles of arrival for legitimate transmissions from a given node [16]. In this way, the receiver can detect an attacker by comparing the angle of arrival of its transmissions against the distribution of previously collected angles of arrival. The matching evaluation metric is the Mahalanobis distance. However, the work does not consider the case of a more powerful attacker who can craft transmitted signals to change the estimated angle of arrival at the receiver.

Aman et al. evaluate the capacity of underwater channels under impersonation attacks [17], assuming that the legitimate receiver uses distance as a feature to discriminate between a legitimate and an impersonating transmitter. After modeling the dynamics of the communications as a Markov chain, the authors numerically optimize the optimum transmission rate for the legitimate transmitter and show that a small neural network reproduces the optimization process well.

In [18], the authors propose to authenticate nodes based on a single feature, the maximum time-reversal resonating strength, which measures how well a received channel impulse response matches those of previous transmissions, stored in a pre-collected database. The authentication mechanism is then based on the Neyman-Pearson likelihood ratio test (LRT).

The approach in [19] considers a large dataset of underwater channel feature measurements and evaluates which features remain coherent over time while becoming uncorrelated already over short distances. Assuming that several trusted nodes hear both the transmissions of a legitimate node and those of an attacker, the proposed method trains generalized Gaussian probability density functions (PDFs) to represent the features of legitimate transmissions. Then, it coordinates the trusted nodes to decide on whether an incoming transmission obeys the previously learned statistics or not. The method is robust against attackers that can precode their transmission to change the channel perceived by multiple trusted nodes at the same time. The above work was further extended in [3] to automatically extract the feature statistics using a neural network, thereby avoiding the need to fit a generalized Gaussian PDF to the channel data. Additionally, in [20] both local training and global training solutions are considered. With local training, each trusted node makes a local decision on authenticity, and a sink uses a neural network to fuse the decisions, making it unnecessary to communicate anything other than the local decisions. Conversely, global training achieves better performance but requires all trusted nodes to communicate the weights of their local neural networks.

In contrast with the existing literature, we do not directly exploit channel impulse response features to tell apart a legitimate node from an attacker in our physical layer authentication approach. Rather, we deploy an LSTM, which tracks the evolution of the chosen set of authentication features and uses the prediction error to discriminate between legitimate transmitter and attacker. This approach factors in mobility by design.

3. System model

We consider three agents: a (legitimate) transmitter, a receiver, and an attacker. The transmitter is mobile, e.g., an AUV, and periodically transmits information to the receiver, via underwater acoustic channels. The receiver can rely on a set of N sensors, $\{S_1, \dots, S_N\}$. the attacker is instead a malicious transmitter that aims to inject fake information into the legitimate by impersonating the legitimate transmitter, sensing proper signal via the underwater acoustic channel.

In turn, the aim of the receiver is then to exploit his N sensors, to decide whether a received packet comes from the legitimate transmitter or the attacker. We assume the channel between the sensor S_n and the actual receiver to be error-free, authenticated, and integrity-protected. Hence, the attacker cannot interfere with the collection of data from the sensors to the logic making the authenticity decision (see Section 4). We assume the sensors to be (at least loosely) synchronized. Considering for instance a scenario where the sensors and The receiver are close to each other, this may be achieved using a wired link. Alternatively, it could be possible to synchronize the sensors using one of the many schemes designed for underwater networks, e.g., [21].

In these conditions, the attacker impersonates the legitimate transmitter by crafting signals with features similar to those of the legitimate transmissions. We assume that the attacker has the advantage of knowing all the details and parameters of the authentication algorithm and that they are also synchronized with the legitimate transmitter and receiver. Moreover, the attacker can precode the transmissions to reproduce any desired channel impulse response at any of the sensors. We remark that these capabilities imply perfect knowledge of the environment, e.g., the surface/bottom profile, as well as the sound speed profile in the network area, and require channel estimation, precoding, and the availability of multiple transceivers. Finally, they also require considerable processing power to compute multiple ray tracing outputs within a negligible amount of time. Thus, the scenario we are considering is quite generous towards the attacker. Still, we assume that the attacker does not know the exact location of the legitimate transmitter, as the attacker can localize the transmitter using the well-known approaches of [22], or matched field processing techniques [23,24] that have limited accuracy. Thus, the attacker cannot track the transmitter's movements accurately. We model the attacker's estimate of the receiver's 3D location vector as

$$\hat{\boldsymbol{P}}_{\mathrm{Tx}} = \boldsymbol{P}_{\mathrm{Tx}} + \boldsymbol{\epsilon}\,, \tag{1}$$

where $\boldsymbol{P}_{\mathrm{Tx}}$ is the true location of the legitimate transmitter and $\boldsymbol{\epsilon}$ models the localization error.

We consider an attacker employing three different strategies, with increasing complexity, from time t_{att},

Type I : the attacker is a malicious static transmitter, sending signals to the sensors.

Type II : the attacker is a moving transmitter. During the attack, they transmit signals while moving.

Type III : the attacker estimates the legitimate transmitter's location and exploits their knowledge of the attacker-sensors channels to pre-compensate the channels.

We remark that, to the best of the authors' knowledge, except for trivial scenarios (e.g., leading to single-arrival channel impulse responses), it is practically impossible to perfectly implement type-III attacks. Hence, in a worst-case analysis fashion, we will also consider type-III attacks as a lower bound on the performance of our authentication scheme. We also remark that obtaining the position of the transmitter and tracking its movements requires an additional effort by the attacker: therefore, having a defense that leverages the legitimate user position to authenticate the source strengthens security as it requires more sophisticated attacks.

Finally, we assume that an authenticated dataset is available for training. This dataset can be collected either from previous operations on the field, by relying on higher-layer authentication mechanisms to act as a bootstrap for the protocol, or from simulation, e.g., by using a ray tracer such as the Bellhop software [25], fed with high-resolution environmental parameters.

4. Proposed protocol

We propose an authentication protocol composed of three steps. After a training phase where each sensor trained its own predictor (more details in Section 4.2), at time t:

1. **Feature Extraction:** each sensor, S_n, estimates the power-delay profile $\{\Pi_n(t,\tau)\}$, where $\Pi_n(t,\tau)$ is the power of the tap with delay τ. Next, it processes the profile to extract the feature vector $\boldsymbol{x}_n(t)$;

2. **Prediction Strategy:** sensor S_n computes

$$\boldsymbol{\delta}_n(t) = \hat{\boldsymbol{x}}_n(t) - \boldsymbol{x}_n(t)\,, \tag{2}$$

where $\hat{x}_n(t)$ is the output of the trained predictor.

3. **Authenticity Verification:** The receiver collects all prediction errors. Additionally, to improve the performance of the scheme, it can collect W observations per sensor, concatenated into a vector

$$\boldsymbol{\delta}(t) = [\delta_1(t),\dots,\delta_1(t-WT),\dots,\delta_N(t)\cdots,\delta_N(t-WT)]\,, \tag{3}$$

where for simplicity we assumed that the signals are collected with a sampling period T. Finally, the receiver computes the decision variable $\gamma = g(\boldsymbol{\delta}(t))$, and tests the authenticity of the packet as

$$\hat{\mathcal{H}} = \begin{cases} 0, & \text{if } \gamma < \lambda \quad \text{(packet from legitimate transmitter)},\\ 1, & \text{if } \gamma \geq \lambda \quad \text{(packet from the attacker.} \end{cases} \tag{4}$$

The threshold λ is chosen a priori by the legitimate user. Let us call $\mathcal{H} = 0$ and $\mathcal{H} = 1$ the case where the legitimate transmitter or the attacker is sending signals, respectively. The false alarm (FA) probability is

$$p_{\text{FA}} = P[\hat{\mathcal{H}} = 1 | \mathcal{H} = 0]\,, \tag{5}$$

while the missed detection (MD) probability is

$$p_{\text{MD}} = P[\hat{\mathcal{H}} = 0 | \mathcal{H} = 1]\,. \tag{6}$$

The threshold λ in (4) can be chosen to meet certain design criteria. However, as feature statistics are typically available only for the legitimate scenario, it is customary for the user to decide on a false alarm probability p_{FA}, and pick as λ the threshold value that yields this probability. During the tests, it is possible to model a specific attack and measure also the p_{MD}. The trade-off between p_{FA} and p_{MD} is typically evaluated by computing the detection error tradeoff (DET) curves.

Following the considerations in [20], the best solution would require us to jointly train both the local predictors and the verification function $g(\cdot)$. Still, in practical circumstances, this is not a viable option as it may be too computationally demanding for UWAN devices [20]. Thus, we resort to a divide-and-conquer approach, where we first process the channel features and later send only the prediction errors to the receiver.

Finally, we remark that, in principle, a federated learning-based solution may speed up the training of the local predictors. Yet, the statistical distribution of the features may vary significantly across the sensors due to the very limited space correlation of underwater acoustic channels. Hence the effort to adapt the online model to the local predictor could be equivalent to the one required when training the predictors from scratch.

In the next Sections, we detail how steps 1–3 are implemented (see Fig. 1).

4.1. Feature extraction

Sensor S_n extracts from the power delay profile $\{\Pi_n(t,\tau)\}$ a set of authentication features, $\boldsymbol{x}_n(t) = [x_n^{(1)}(t),\dots,x_n^{(5)}(t)]$, where $x_n^{(i)}(t)$ is the measurement of feature i, collected at time t by sensor S_n.

To avoid unnecessary computations, we first zero out low-power arrivals in the power-delay profile as

$$\Pi'_n(t,\tau) = \begin{cases} 0, & \text{if } \Pi_n(t,\tau) < T_h,\\ \Pi_n(t,\tau), & \text{if } \Pi_n(t,\tau) \geq T_h. \end{cases} \tag{7}$$

where threshold $T_h = 10^{-8}\max_\tau |\Pi_n(t,\tau)|$ trims exceedingly low-power arrivals that would be buried in noise.[1]

$\mathcal{H}_n(t)$ is then the set of delays of all channel arrivals that remain after thresholding.

Concerning the actual set of features, we combine the set used in [3,19] with the power-weighed average delay proposed instead in [4]. In detail, we consider the channel impulse response features defined in the following:

Number of channel taps, which hints at the spread of the acoustic channel

$$x_n^{(1)}(t) = |\mathcal{H}_n(t)|\,. \tag{8}$$

Average tap power,

$$x_n^{(2)}(t) = \frac{1}{|\mathcal{H}_n(t)|}\sum_{\tau\in\mathcal{H}_n(t)} |\Pi'_n(t,\tau)|\,, \tag{9}$$

Relative RMS delay, which reflects the delay spread of the channel,

$$x_n^{(3)}(t) = \left(\frac{1}{|\mathcal{H}_n(t)|-1}\sum_{\tau\in\mathcal{H}_n(t),\tau\neq\tau_0}(\tau-\tau_0)^2\right)^{1/2}, \tag{10}$$

where τ_0 is the delay of the first tap, i.e., $\tau_0 = \min\{\tau : \tau \in \mathcal{H}_n(t)\}$.

Smoothed received power, that tracks the variation of power over time. In particular, let $q_{n,t}$ be the power of a symbol received by node n at time instance t, and a user-defined parameter $0 \leq \alpha \leq 1$, thus

$$x_n^{(4)}(t) = \alpha\, q_{n,t} + (1-\alpha)\, x_n^{(4)}(t-T)\,, \tag{11}$$

where $x_n^{(4)}(t-T)$ is the smoothed received power of the previous symbol, received at time $t-T$.

Power-weighed average delay

$$x_n^{(5)} = \frac{1}{\bar{\Pi}_n(t)}\sum_{\tau\in\mathcal{H}_n(t)} \tau\Pi'_n(t,\tau)\,, \tag{12}$$

where, as pointed out in Section 3, we assume all devices to be synchronized, with

$$\bar{\Pi}_n(t) = \sum_{\tau\in\mathcal{H}_n(t)} \Pi'_n(t,\tau)\,. \tag{13}$$

Notice that the delay of the first arrival is strongly correlated to the distance between the transmitting and receiving devices.

[1] Alternatively, it would be possible to fix a predefined FA probability for the discrimination of true vs. noise-induced peaks in the channel impulse response and compute the threshold leading to such FA probability according to the equations in [26].

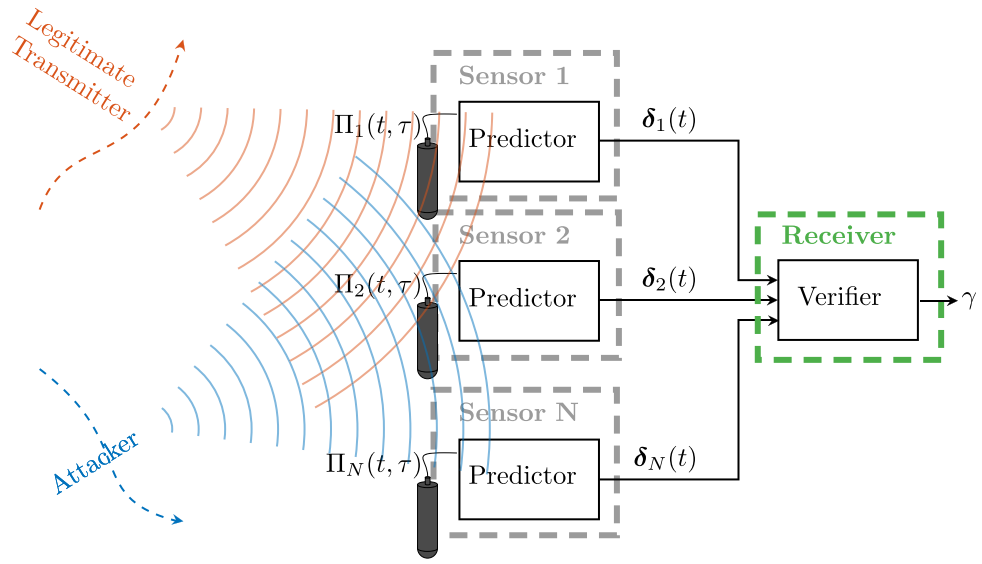


Fig. 1. High-level sketch of the proposed solution.

The above features have a desirable property: they enable a good separation among channel responses generated by almost-stationary transmitters located at different positions [19] and they relate well with distance [4]. It remains possible to extend the feature set, e.g., by considering the features of [27].

4.2. Prediction strategy

In this Section, we describe both the prediction strategy and the predictor training. To track the evolution over time of the authentication feature vector $\boldsymbol{x}(t)$, each sensor exploits a LSTM RNN. Before introducing LSTMs we briefly review deep feedforward neural networks (NNs). Deep NNs are composed of a series of layers, each possibly containing several neuron units, yielding the so-called *multi-layer perceptron* model. In particular, considering a NN composed of Q layers, the network is said to be feedforward if the input of any neuron of the $(q+1)$-th layer is the collection of the output of the neurons in the qth layer. Additionally, the NN is fully connected if the input of each neuron at layer $q+1$ is a weighed sum of the outputs from all neurons of layer q. This is in contrast to other NN architectures, such as convolutional NNs.

Each neuron is associated to a weight $\boldsymbol{w}_k^{(q)}$, a bias $b_k^{(q)}$, and an activation function $f^{(q)}(\cdot)$. Therefore, considering a feed-forward fully connected NN, the output of the kth neuron of the qth layer is computed as

$$y_k^{(q+1)} = f^{(q)}\left(\boldsymbol{w}_k^{(q)}\boldsymbol{y}^{(q)} + b_k^{(q)}\right). \tag{14}$$

To implement a predictor, we consider a so-called *regression* task, where the network has to compute the future feature from the past one. In more detail, we model the predictor as a function f, the output of the last layer for input $\boldsymbol{x}$ is $\boldsymbol{x}'_n = \boldsymbol{y}^{(Q)} = f(\boldsymbol{x})$. First, we model the network architecture such that $\boldsymbol{y}^{(Q)}$ has the same size as the input. The training loss will then be the mean square error (MSE) loss

$$\ell = \left\| f\left(\boldsymbol{x}_n(t-T)\right) - \boldsymbol{x}_n(t)\right\|^2. \tag{15}$$

The NN is trained (i.e., optimized) using algorithms such as the adaptive moment estimation (ADAM), by setting as target $\{\boldsymbol{x}_n\}$ for input the corresponding $\{\boldsymbol{x}'_n\}$. The main issue of multilayer perceptron NNs is that they have no memory buffer, and are thus unable to track the evolution of the feature vector over time.

An alternative candidate to NNs is a Kalman filter, as we have previously proposed in [4]. However, the Kalman filter requires knowledge about both the state-transition and the evolution models [28, Ch. 13]. Different from the power-weighed average delay, it is hard to analytically relate the authentication features to the relative motion between the transmitter and the receiver. We resort instead to RNNs, focusing on LSTMs.

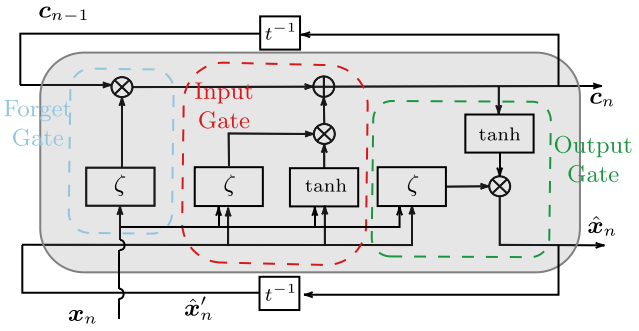


Fig. 2. Scheme for the LSTM cell used to implement the predictor.

LSTM are provided with a set of cell states, where $\boldsymbol{c}_n$ is the state at time t_n. By exploiting the cell state, RNNs can learn the correlation, and thus the model governing the evolution of subsequent samples of a given metric over time. In detail, the architecture can be split into three gates, the *forget*, the *input*, and the *output* gate. The forget gate aims to properly weigh the influence of the (past) cell state on the new cell state and the output, i.e., how much information should be forgotten or maintained between one input and the next one. The input gate takes in the input and updates the cell state, considering both the current input, the output of the forget gate, and the previous output. Finally, the output gate computes the actual output from the updated cell state and the current input. The training procedure for RNNs is identical to the one for the NNs, but all cell states are reset after each training sequence ends. More details about both NNs and RNNs can be found in [29, Ch. 6,9].

A sketch of an LSTM cell is reported in Fig. 2, where $\tanh(\cdot)$ and $\zeta(\cdot)$ are the hyperbolic tangent and sigmoid activation functions.

Finally, each sensor computes the prediction error $\boldsymbol{\delta}_n(t)$ as in (2) and transmits it to the receiver.

4.3. Authenticity verification

In this section, we detail how the receiver exploits the prediction error $\boldsymbol{\delta}(t)$, which collects the n prediction error vectors $\boldsymbol{\delta}_n(t)$ from the n

sensors, to compute the test variable γ. This will be in turn exploited to verify the authenticity of the transmitter by using the test (4).

As mentioned in Section 3, we frame the problem using binary hypothesis testing, considering $\mathcal{H}_0$ and $\mathcal{H}_1$ as the legitimate and the under-attack scenario, respectively. In other terms, the problem is then to distinguish between $\boldsymbol{\delta}(t) \in \mathcal{H}_0$ and $\boldsymbol{\delta}(t) \in \mathcal{H}_1$.

Given $\boldsymbol{\delta}$, the optimal test, i.e., the test minimizing the p_{md} for a fixed false alarm p_{FA}, is the LRT

$$\mathcal{L}(\boldsymbol{\delta}) = \frac{p(\boldsymbol{\delta}|\mathcal{H}_0)}{p(\boldsymbol{\delta}|\mathcal{H}_1)} \gtrless \varphi \,. \tag{16}$$

However, this test cannot be used in our scenario, as it requires the distributions of $\boldsymbol{\delta}(t)$ in both the legitimate and the under-attack scenarios. These distributions are not available, and cannot be estimated in real-time. Additionally, even assuming the availability of the first $p(\boldsymbol{\delta}|\mathcal{H}_0)$, e.g., by running (as accurate as possible) simulations and measuring the distribution from the dataset, the latter would require a detailed attack model, and thus also the knowledge of the specific attack chosen by the attacker.

As a common practice [30], we resort to the GLRT

$$\mathcal{G}(\boldsymbol{\delta}) = p(\boldsymbol{\delta}|\mathcal{H}_0) \gtrless \varphi \,. \tag{17}$$

It is worth pointing out, that given enough training data, it is possible to implement such a test even using data-driven solutions, such as the one-class least-squares support vector machines [31].

To model the statistical distribution of the prediction error, we consider a Gaussian overbound. In particular, we assume the prediction error components to be iid and Gaussian-distributed under $\mathcal{H}_0$, where the ith component has distribution $\delta_i(t) \sim \mathcal{N}(0, \sigma_i^2)$ and $\boldsymbol{\delta}(t) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$. Hence, for a prediction error vector of size K, for the GLRT (17), it yields

$$\mathcal{G}(\boldsymbol{\delta}) = \frac{1}{(2\pi)^K \sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}\boldsymbol{\delta}^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} \boldsymbol{\delta}\right) \gtrless \varphi \,, \tag{18}$$

which is equivalent[2] to

$$\mathcal{G}'(\boldsymbol{\delta}) = \sum_{i=1}^{K} \frac{\delta_i^2}{\sigma_i^2} \lessgtr \varphi' \,. \tag{19}$$

We consider the following remarks. First, we see that such an operation is essentially a normalization of each error term, leading to a sum of squared standard normal variables (i.e., chi-square random variables). Moreover, we remark that each variance σ_i^2 can be either estimated a priori or from a dataset.

The GLRT is a sum: thus, instead of summing up all the (normalized) terms at the receiver, it becomes possible to compute partial sums at the sensors and transmit only these sums, reducing the amount of communicated data.

The Gaussian hypothesis is reinforced when combining the prediction error from multiple sensors and multiple instants of time, by the central limit theorem. In Section 5, we will confirm that this assumption is effective, as it enables us to detect the attacks compactly.

5. Numerical results

In this Section, we test the performance of the proposed RNN-based authentication strategy. We first describe the simulation scenario and modeling. Next, we check the performance of the predictor, by verifying that, when under attack, different behavior is observed, which can be used to distinguish between $\mathcal{H}_0$ and $\mathcal{H}_1$. Finally, we test the performance of the overall scheme against all the considered attack classes.

[2] It requires a remapping of the thresholds but achieves the same p_{FA} and p_{MD}.

5.1. Simulation scenario

We evaluate the proposed approach by simulating underwater acoustic communication channels via Bellhop [25]. In more detail, we considered a region of the San Diego bay area (32°52′34.5″N, 117°24′12.8″W) of size equal to about 6 km × 6 km, having a depth between 250 and 650 m.

At the start of each simulation, we randomly deploy the attacker and the receiver within the area. In particular, the receiver incorporates four sensors arranged as a tetrahedral pyramid of base radius and height equal to 5 m.

We consider the legitimate transmitter to be a moving device transmitting acoustic signals in broadcast, once every $\Delta t = 1$ s. the legitimate transmitter moves across the area according to a correlated Gauss–Markov mobility model, starting at a random location, $\boldsymbol{P}_{\text{A},0}$, with an initial velocity vector of magnitude $v_0 = \|\boldsymbol{v}_{\text{A},0}\|$ and direction drawn uniformly at random in an interval of 45° around due north. Once every Δt, the legitimate transmitter's location $\boldsymbol{P}_{\text{A},i}$ and velocity $\boldsymbol{v}_{\text{A},i}$ are updated from step t_{i-1} to t_i, as

$$\boldsymbol{P}_{\text{A},i} = \boldsymbol{P}_{\text{A},i-1} + \boldsymbol{v}_{\text{A},i}\Delta t, \tag{20a}$$

$$\boldsymbol{v}_{\text{A},i} = \alpha\, \boldsymbol{v}_{\text{A},i-1} + \boldsymbol{\eta}\sqrt{1-\alpha^2}\,, \tag{20b}$$

where i and $i-1$ refer to the current and previous location and velocity update epochs, respectively, $\alpha = 1 - 2 \cdot 10^{-3}$ is the trajectory correlation factor, and $\boldsymbol{\eta}$ is a Gaussian noise vector having (fixed) independent components of standard deviation $[2, 2, 1]$ m/s along the east–west, north–south and depth dimensions, respectively. These choices lead to correlated trajectories, which reproduce the uncertainty of drifting due to currents and eddies. The lower variance along the depth dimension models the typically more accurate depth-keeping capability of underwater mobile devices. Fig. 3 shows the bathymetry map of the area, the locations of the sensors, the attacker, and the legitimate transmitter's trajectory for one instance of our simulations.

We run a Monte-Carlo simulation, including several realizations of the above scenario, with different initial random locations and trajectories. Additionally, we tested different initial velocity magnitudes, with $v_0 = 0.5, 1$, and 1.5 m/s. In total, we generated 20 simulations for each initial velocity magnitude v_0; each simulation lasts 12 000 s, corresponding to a total of 12 000 power-delay profiles collected per simulation.

We assume that, for each simulation, there is an initial training period when each sensor S_n receives data only from the legitimate transmitter. In each simulation, each sensor collects a total of 12 000 (legitimate) feature vectors, and uses the first 30% to train the neural networks. The rest of the data is then used as a test set. Concerning the features' parameters, after manual tuning we set the smoothing received power coefficient (11) to $\alpha = 0.5$.

We implemented the attacks described in Section 3. In particular, for type III attacks, we note that it is not feasible to pre-compensate all channels towards all the sensors at the same time, even with dedicated hardware. Thus, even if the attacker can perfectly estimate the channel towards sensor i, this will still introduce errors to the other spoofed channels. We modeled this effect by translating the attacker's uncertainty about the legitimate transmitter's location into an uncertainty on the channel that should be reproduced by the attacker to impersonate the legitimate transmitter successfully at each sensor.

In more detail, we displace the legitimate transmitter uniformly at random within a radius of either $\epsilon = \{50, 100, 200\}$ m over the azimuthal plane, and within a depth of ±20 m from the legitimate transmitter's actual location. These values are representative of realistic errors obtained via localization schemes based on matched field processing [23]. For each random displacement, we recompute the channel impulse response at each of the sensors.

Finally concerning the LSTM architectures, we designed the RNN to be lightweight, thus composed only of two layers: a LSTM layer with 10 cells, and a fully connected layer with 5 neurons.

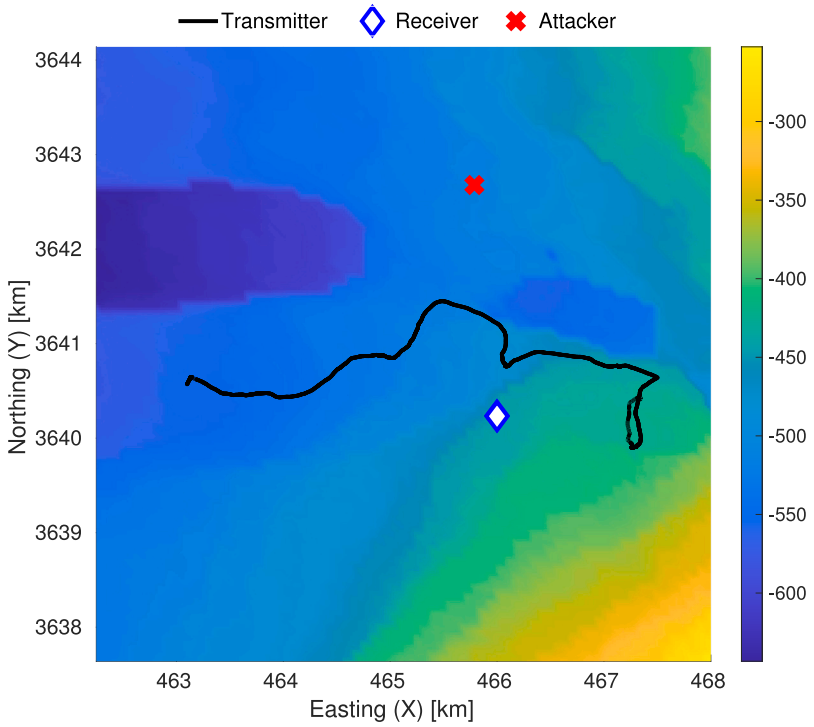


Fig. 3. Example of simulation scenario showing the location of the attacker and the sensors, and a sample trajectory for the legitimate transmitter. The background colors convey the local depth. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2. Single sensor prediction

First, we present the results of the prediction strategy considering only sensor S_1 for both a legitimate and an under-attack scenario, to show that the predictors exhibit an anomalous behavior when under attack.

In particular, we consider a type III attack, where the attacker can replicate the legitimate transmitter channel with a localization error $\epsilon = 100$ m. Fig. 4 reports the predictions for each channel-related metric in both the legitimate and the under-attack cases, compared with the ground truth. The attack starts at $t_{\text{att}} = 1000$. First, we notice that the predictors can correctly track the features' behavior over time, as no relevant errors between predicted and ground truth values are shown. Next, anomalous behaviors are exhibited by the predictor when the attack starts, as noticeable increases of the predictor's fluctuations are observed after t_{att}. Indeed, this shows that our predictor can be used for authentication purposes. The smoothed received power and power-weighted delay perform better in terms of security, as they have a clear bias to the ground truth. On the other hand, these features also present more errors with respect to the ground truth, which may lead to higher false alarms, with respect to the other three. Thus, we can conclude that there is no advantage and, as long as the receiver does not have limitation on the number of features to be tracked, the best solution is to include as many features as possible.

5.3. Authentication verification

Here, we report the results of the authentication procedure testing the effectiveness of the proposed procedure in both legitimate and under attack.

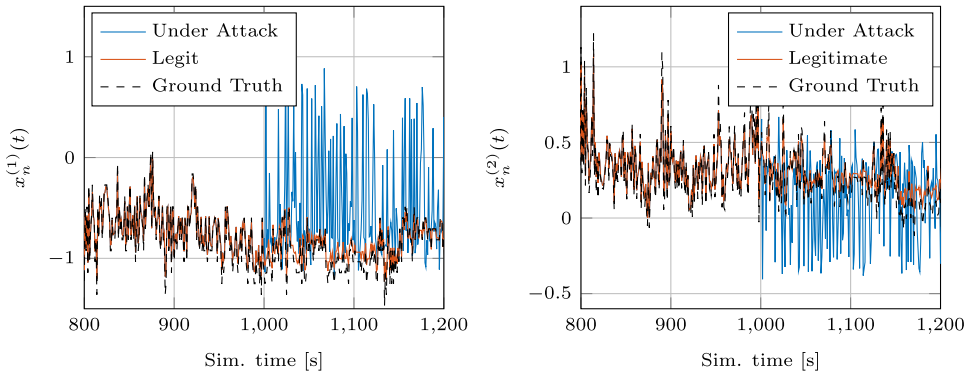
First, about the prediction results of Fig. 4, we want to check whether the prediction error is a valid metric to distinguish legitimate transmissions from fake ones. Fig. 5 reports the cumulative distribution function (CDF) the norms of the normalized predictions error measured at each sensor, considering a window of size $W = 1$, and the Type I attack. We draw two key observations. First, it may be possible to effectively distinguish the attacker from the legitimate transmissions at the local level. Next, there may be some fluctuations in the performance among different sensors: for instance, in this simulation, the best performance is achieved by sensor 4. So we can conclude that there are in general benefits in sharing the prediction errors between the sensors as, in the worst case, all the sensors could achieve the performance of the best one.

Next, we evaluate the data aggregation at the receiver and the GLRT performance, in terms of DET curves, measuring the p_{MD} as a function of p_{FA}, for all the attacks described in Section 3 and different scenario parameters. As a means of comparison, we consider an authentication scheme inspired by [19]. Namely, we consider a model where the verifier uses part of the dataset to estimate the feature distribution and then performs a GLRT on the measurement to be tested to verify the authenticity of a message. In more detail, each sensor exploits the training dataset to estimate the distribution pdf of each feature in the legitimate case, $\hat{p}_i(x|\mathcal{H}_0)$, by using kernel density estimation (KDE). Next, assuming all the features to be statistically independent, the authenticity of measured channel $\boldsymbol{x}_n$ is verified by computing the (local) GLRT

$$\mathcal{G}'(\boldsymbol{x}_n) = \sum_{i=1}^{5} \log \hat{p}_i(\boldsymbol{x}_n|\mathcal{H}_0) \gtrless \varphi' \,. \tag{21}$$

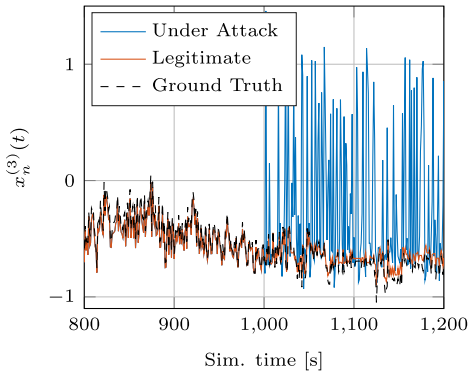
To test the performance of the KDE-based scheme we ran 100 simulations: on each simulation, we used the first 3000 measurements contained in the training dataset to estimate each pdf $\hat{p}_i(x|\mathcal{H}_0)$. Concerning the testing dataset, the legitimate part includes 1000 measurements subsequent to the ones of the training dataset, whereas, for the under-attack case, we considered the 1000 measurement collected after an attack of Type I.

Fig. 6 reports the DET curves obtained using the KDE-based check compared with a guessing check, where the receiver randomly decides whether the measurement is legitimate or not. Indeed the naive check is not a viable option as the performance is almost comparable to the guessing check. This is because the measurements $\boldsymbol{x}_n$ can be associated with a (highly) non-stationary random process, thus the distribution

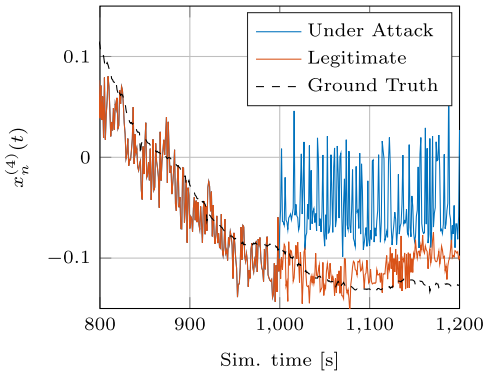


(a) Number of Channel Taps.

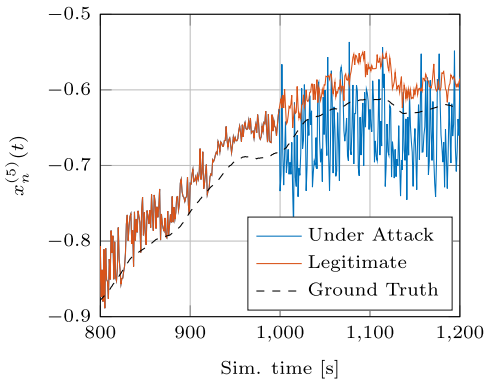
(b) Avg. Tap Power.



(c) RMS Delay.



(d) Smoothed RX.



(e) Power-weighted Average Delay.

Fig. 4. Predicted features, $\hat{x}_n^{(i)}(t)$, in a scenario with only legitimate transmissions (orange) compared to a scenario where an attack of Type III with $\epsilon = 100\,\text{m}$, taking place after 1000 s of operation (blue), compared to the ground truth measurement $x_n^{(i)}(t)$ (black, dashed). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

measured on the first part of the dataset is different from the one associated with the second part. This motivates us to look for solutions that do not rely on the actual feature distribution but on the (coherence with) the feature evolution over time. Additionally, we notice that the decision locally taken by each sensor does not bear relevant information, therefore even collecting the decisions from multiple sensors will not allow the receiver to distinguish between attacker and legitimate transmitter effectively.

Fig. 7 reports instead the results with $W = 10$ for $v_0 = 1\,\text{m/s}$, for the attacks of type I, II, and III when using the proposed check. In more detail, we consider two cases for the Type III attack, where the attacker is characterized by channel reproduction accuracies A) $\epsilon_\text{A} = [0, 50, 50, 100]\,\text{m}$, or B) $\epsilon_\text{B} = [0, 100, 200, 200]\,\text{m}$. For example, in case A), this means that the attacker can perfectly reproduce the first feature while reproducing the remaining features with an error as if the attacker estimated the location of the legitimate transmitter within a radius of respectively 50, 50, and 100 m from the actual location. As expected, the proposed attack achieves the best performance against the type I attack, followed by the II and III. Concerning the type III attacks, scenario A) results are harder than those of scenario B), as we are assuming that fewer errors are introduced by the attacker. Still, even in this worst-case analysis, it is possible to distinguish between the legitimate transmitter's and the attacker's signals.

Fig. 8 reports instead the results obtained for the attack of type II, considering $W = 10$ but training the network on either the same trajectory of the test or on a different trajectory. We observe that, while the first case expectedly achieves slightly better results, good performance is still achieved even in the latter case. This confirms

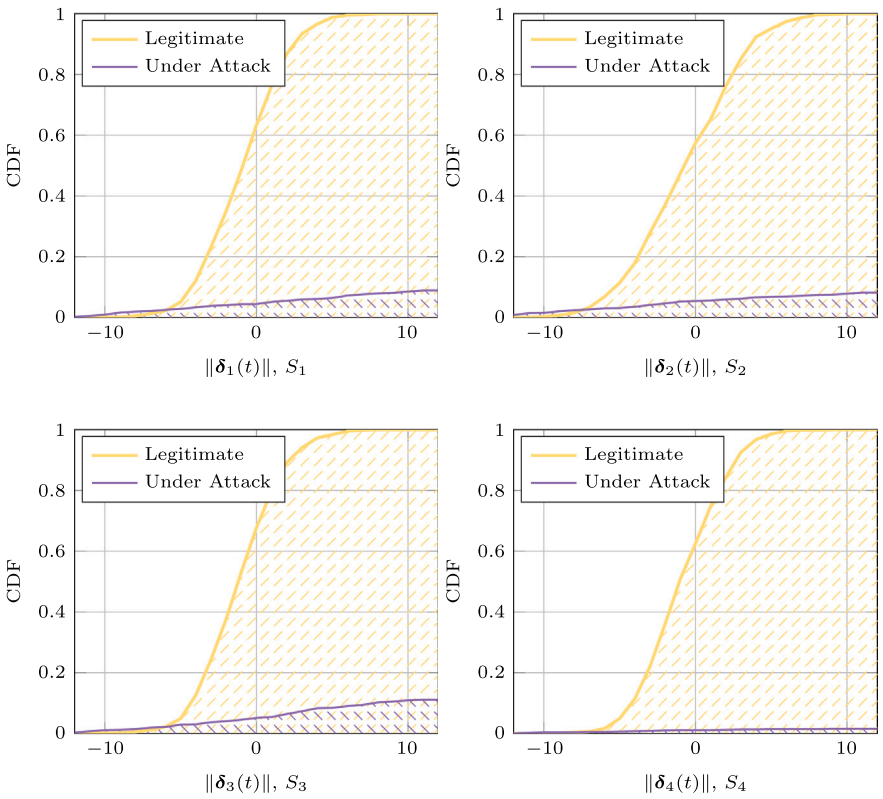


Fig. 5. CDF of the norm of the prediction errors $\boldsymbol{\delta}_n(t)$, measured at different receivers, in the legitimate (yellow) and under attack scenarios (purple). The x-axis has been limited for graphical purposes, as the CDF for the under-attack case converges to 1 at $x \gg 12$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

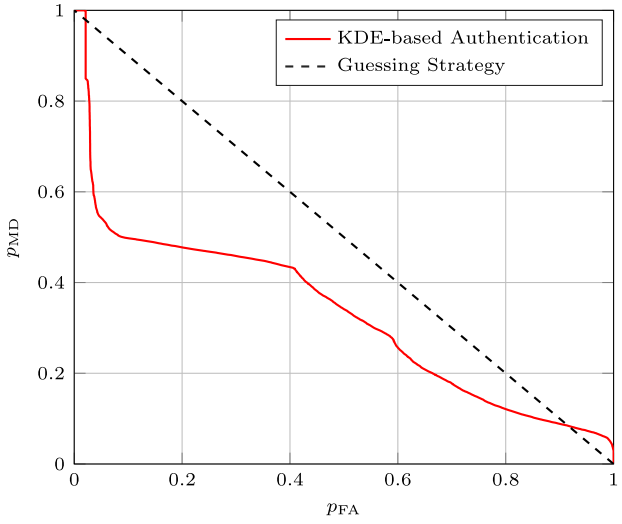


Fig. 6. DET curves for as single sensors against the Type I attack. Comparison between KDE-based check (red) and guessing strategy (red, dashed). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

that the neural network correctly learns the statistical properties of the authentication features (which exhibit highly different patterns in the presence of an attack, see Fig. 4), rather than specializing in a specific trajectory. This could be a solution to a scenario where we have limited training data, as it means that it could be possible to train the networks on simulated data and then either directly use that on deployed sensors at sea, or to have offline training and, after the actual deployment, a short online training to refine the networks.

Fig. 9 reports instead the results focusing on the (easier to launch) Type-I attack but considering as legitimate receiver velocities $v_0 = 0.5$, 1, and 1.5 m/s. While no relevant performance change is observed for $v_0 = 0.5$ and 1 m/s, a slightly worse performance is achieved for $v_0 = 1.5$ m/s. Still, we expect the algorithms to perform worse for high movement velocity, as the channel becomes more unstable and difficult to predict. As the prediction error increases, there starts to be room for the attacker to launch a successful attack.

6. Conclusions

We proposed a physical layer-based authentication algorithm for UWAN where the legitimate transmitter is a mobile device. The receiver employs a set of sensors equipped with a previously trained LSTM network to track the evolution of the considered channel feature set. Next, the receiver collects all prediction errors and then exploits them through the GLRT to decide whether the signal is legitimate or fake, i.e., transmitted by the legitimate transmitter or by the attacker.

Numerical results have been obtained using the Bellhop ray-tracing software. A first batch of results confirms that the LSTMs is (*i*) able to correctly track the channel features over time and (*ii*) exhibits anomalous behavior when the attack starts. This confirms that LSTMs can be used for authentication purposes even in the challenging scenario where the attacker can impersonate the legitimate transmitter at one of the sensors.

Next, we tested the performance of the scheme in different attack scenarios, with different levels of feasibility, and evaluated the resulting DET curve. Our results prove that the proposed protocol can detect both naive attacks, and more sophisticated attacks feasible only when the attacker has many more resources than the legitimate devices.

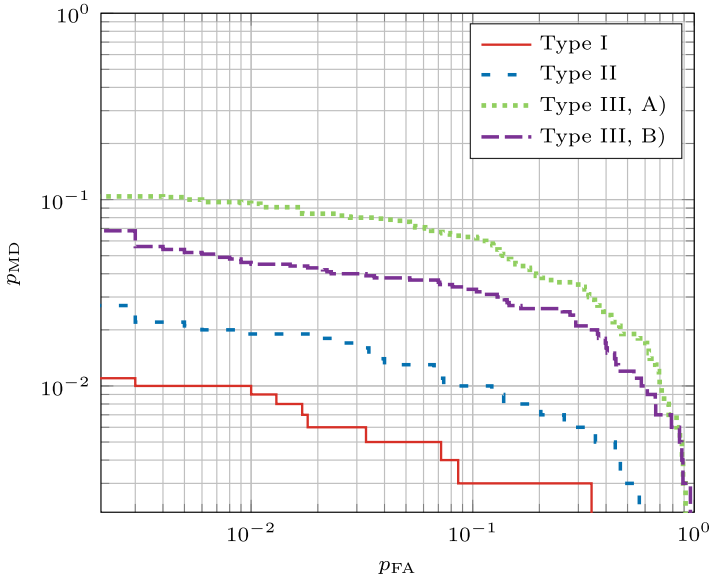


Fig. 7. DET curves for the Type I, II, and III attacks, for a velocity $v_0 = 1\,\mathrm{m/s}$, with $W = 10$ samples.

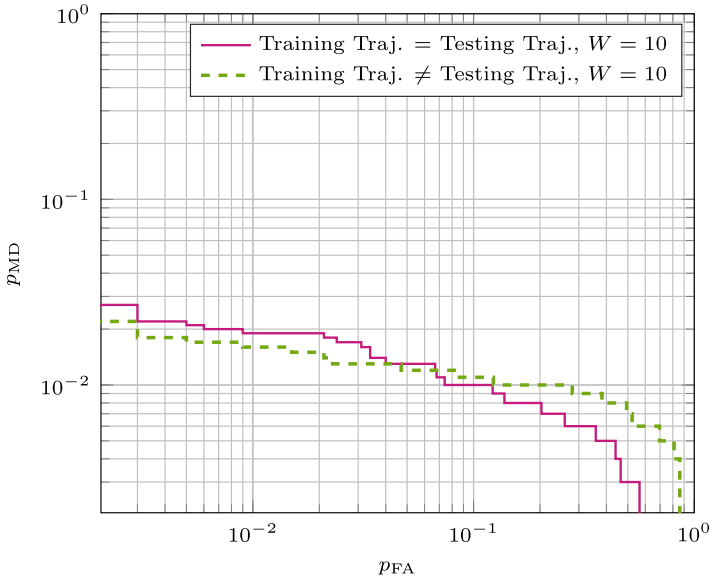


Fig. 8. DET curves for the Type II attack, for a velocity $v_0 = 1\,\mathrm{m/s}$, with $W = 10$ samples using a network trained on (part of the) testing dataset, and on a different trajectory.

CRediT authorship contribution statement

Francesco Ardizzon: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Paolo Casari:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Stefano Tomasin:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Francesco Ardizzon reports financial support was provided by NATO. Paolo Casari reports financial support was provided by NATO. Stefano Tomasin reports financial support was provided by NATO. Ardizzon Francesco reports financial support was provided by European Union. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was sponsored in part by the NATO Science for Peace and Security Programme under grant no. G5884 (SAFE-UComm), and by the Italian Ministry of Economy and Finance through iNEST (Interconnected NordEst Innovation Ecosystem), funded by PNRR (Mission 4.2, Investment 1.5), Next Generation EU (Project ID: ECS 00000043, Digital, Industry, Aerospace).

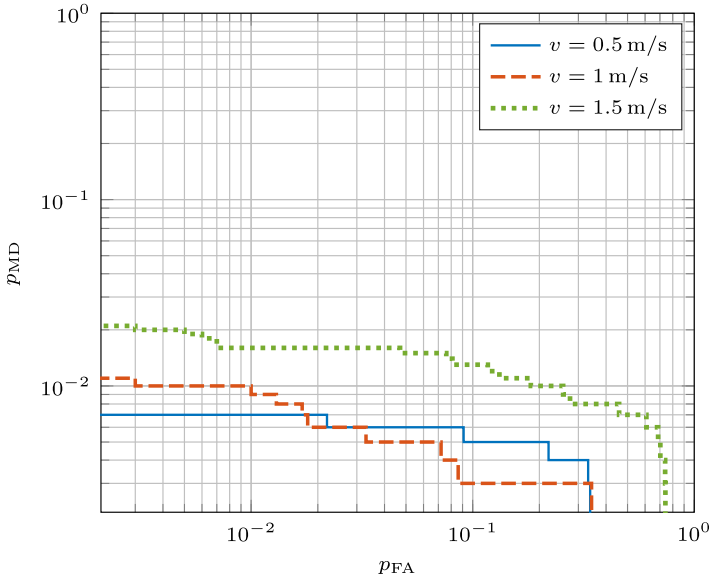


Fig. 9. DET curves for the Type I attack, for velocities $v_0 = 0.5$, 1, and 1.5 m/s, with $W = 10$ samples.

References

[1] M. Stojanovic, J. Preisig, Underwater acoustic communication channels: Propagation models and statistical characterization, IEEE Commun. Mag. 47 (1) (2009) 84–89.

[2] E. Jorswieck, S. Tomasin, A. Sezgin, Broadcasting into the uncertainty: Authentication and confidentiality by physical-layer processing, Proc. IEEE 103 (10) (2015) 1702–1724.

[3] L. Bragagnolo, F. Ardizzon, N. Laurenti, P. Casari, R. Diamant, S. Tomasin, Authentication of underwater acoustic transmissions via machine learning techniques, in: Proc. IEEE COMCAS, IEEE, Tel Aviv, Israel, 2021, pp. 255–260.

[4] P. Casari, F. Ardizzon, S. Tomasin, Physical layer authentication in underwater acoustic networks with mobile devices, in: Proc. ACM WUWNet, Boston, MA, USA, 2022, [Online]. Available: https://doi.org/10.1145/3567600.3567604.

[5] W. Aman, S. Al-Kuwari, M. Muzzammil, M.M.U. Rahman, A. Kumar, Security of underwater and air–water wireless communication: State-of-the-art, challenges and outlook, Ad Hoc Netw. 142 (2023) 103–114.

[6] S. Jiang, On securing underwater acoustic networks: A survey, IEEE Commun. Surv. Tutor. 21 (1) (2019) 729–752.

[7] E. Souza, H.C. Wong, I. Cunha, L.F.M. Vieira, L.B. Oliveira, End-to-end authentication in under-water sensor networks, in: Proc. IEEE ISCC, Split, Croatia, 2013, pp. 000299–000304.

[8] G. Dini, A. Lo Duca, A secure communication suite for underwater acoustic sensor networks, Sensors 12 (2012) 15133–15158, [Online]. Available: http://dx.doi.org/10.3390/s121115133.

[9] A. Menezes, P. van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, FL, 1996.

[10] D. Muhammed, M.H. Anisi, M. Zareei, A. Khan, Game theory-based cooperation for underwater acoustic sensor networks: Taxonomy, review, research challenges and directions, Sensors 18 (2) (2018).

[11] M. Sharif-Yazd, M.R. Khosrav, M.K. Moghimi, A survey on underwater acoustic sensor networks: Perspectives on protocol design for signaling, MAC and routing, J. Comput. Commun. 5 (5) (2017) 12–23.

[12] C. Yuan, W. Chen, Y. Zhu, D. Li, J. Tan, A low computational complexity authentication scheme in underwater wireless sensor network, in: Proc. MSN, IEEE, Shenzhen, China, 2015, pp. 116–123.

[13] A. Al Guqhaiman, O. Akanbi, A. Aljaedi, C.E. Chow, Lightweight multi-factor authentication for underwater wireless sensor networks, in: Proc. CSCI, IEEE, Las Vegas, NV, USA, 2020, pp. 188–194.

[14] S. Zhang, X. Du, X. Liu, A secure remote mutual authentication scheme based on chaotic map for underwater acoustic networks, IEEE Access 8 (2020) 48285–48298.

[15] A.A. Islam, K.A. Taher, A novel authentication mechanism for securing underwater wireless sensors from sybil attack, in: Proc. ICEEICT, IEEE, Mirpur, Dhaka, 2021, pp. 1–6.

[16] M. Khalid, R. Zhao, N. Ahmed, Physical layer authentication in line-of-sight underwater acoustic sensor networks, in: Proc. MTS/IEEE OCEANS, IEEE, Singapore, 2020, pp. 1–5.

[17] W. Aman, Z. Haider, S.W.H. Shah, M.M. Ur Rahman, O.A. Dobre, On the effective capacity of an underwater acoustic channel under impersonation attack, in: Proc. IEEE ICC, IEEE, Dublin, Ireland, 2020.

[18] R. Zhao, M. Khalid, O.A. Dobre, X. Wang, Physical layer node authentication in underwater acoustic sensor networks using time-reversal, IEEE Sens. J. 22 (4) (2022) 3796–3809.

[19] R. Diamant, P. Casari, S. Tomasin, Cooperative authentication in underwater acoustic sensor networks, IEEE Trans. Wirel. Commun. 18 (2) (2019) 954–968.

[20] F. Ardizzon, S. Tomasin, R. Diamant, P. Casari, Machine learning-based distributed authentication of UWAN nodes with limited shared information, in: Proc. UCOMMS, Lerici, Italy, 2022.

[21] A. Vermeij, A. Munafò, A robust, opportunistic clock synchronization algorithm for ad hoc underwater acoustic networks, IEEE J. Ocean. Eng. 40 (4) (2015) 841–852.

[22] E. Dubrovinskaya, V. Kebkal, O. Kebkal, K. Kebkal, P. Casari, Underwater localization via wideband direction-of-arrival estimation using acoustic arrays of arbitrary shape, Sensors 20 (14) (2020) 1–20.

[23] E. Dubrovinskaya, P. Casari, R. Diamant, Bathymetry-aided underwater acoustic localization using a single passive receiver, J. Acoust. Soc. Am. 146 (6) (2019) 4774–4789.

[24] Z.-H. Michalopoulou, P. Gerstoft, D. Caviedes-Nozal, Matched field source localization with Gaussian processes, JASA Express Lett. 1 (6) (2021).

[25] M. Porter, et al., Bellhop code, 2018, [Online]. Available: http://oalib.hlsresearch.com/Rays/index.html.

[26] R. Diamant, Closed form analysis of the normalized matched filter with a test case for detection of underwater acoustic signals, IEEE Access 4 (2016) 8225–8235.

[27] G. Sklivanitis, K. Pelekanakis, S. Yıldırım, R. Petroccia, J. Alves, D. Pados, Physical layer security against an informed eavesdropper in underwater acoustic channels: Reconciliation and privacy amplification, in: Proc. UCOMMS, IEEE, 2021, pp. 1–5.

[28] S. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[29] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.

[30] M. Ceccato, F. Formaggio, N. Laurenti, S. Tomasin, Generalized likelihood ratio test for GNSS spoofing detection in devices with IMU, IEEE Trans. Inf. Forensics Secur. 16 (2021) 3496–3509.

[31] F. Ardizzon, S. Tomasin, On the generalized likelihood ratio test and one-class classifiers, 2023, arXiv.

Francesco Ardizzon received the B.Sc. degree in 2016, the M.Sc. degree in 2019, and the Ph.D. degree in Information Engineering in 2023 from the University of Padova, Italy. In 2022 he has been a visiting scientist at the ESA European Space Research and Technology Centre. He is currently an Assistant Professor at the University of Padova. His current research interests include authentication for global navigation satellite systems, physical layer security, and underwater acoustic communications.

Paolo Casari received the Ph.D. degree in information engineering from the University of Padova, Italy, in 2008. He was on leave with the Massachusetts Institute of Technology in 2007, working on underwater communications and networks. In 2015, he joined the IMDEA Networks Institute, Madrid, Spain, where he led the Ubiquitous Wireless Networks Group. In October 2019, he joined the Faculty of the University of Trento, first as a tenure-track Assistant Professor and as an Associate Professor since October 2022. He collaborated to several funded projects, including EU FP7 and H2020 efforts, EDA projects, as well as US ARO, ONR, and NSF initiatives. He is currently a PI of the NATO SPS Project SAFE-UComm. His research interests include diverse aspects of networked communications and computing, such as channel modeling, network protocol design, localization, resource allocation, security, simulation, and experimental evaluation. He is on the editorial boards of the IEEE Transactions on Mobile Computing and the IEEE Transactions on Wireless Communications, and regularly serves on the organizing committee of several international conferences. He received two best paper awards.

Stefano Tomasin received the Ph.D. degree from the University of Padova, Italy, in 2003. He joined the University of Padova where he is now Full Professor (since 2022). He was visiting faculty at Qualcomm, San Diego (CA) in 2004, the Polytechnic University in Brooklyn (NY) in 2007 and the Mathematical and Algorithmic Sciences Laboratory of Huawei in Paris (France) in 2015. His current research interests include physical layer security, security of global navigation satellite systems, signal processing for wireless communications, synchronization, and scheduling of communication resources. He is a senior member of IEEE since 2011 (member since 1999) and a member of EURASIP since 2011. He is a Deputy Editor-in-Chief of the IEEE Transactions on Information Forensics and Security since January 2023.