Journal of Electronic Science and Technology 20 (2022) 100181

Contents lists available at ScienceDirect

Journal of Electronic Science and Technology

journal homepage: www.keaipublishing.com/en/journals/journal-of-electronic-science-and-technology/

Inaudible sound covert channel with anti-jamming capability: Attacks vs. countermeasure

Xiao-Hang Wang [a,b], Shou-Bin Li [c], Ying-Tao Jiang [d], Amit Kumar Singh [e], Bi-Yun Ma [f], Le-Tian Huang [g], Mei Yang [d], Fen Guo [c,*]

[a] School of Cyber Science and Technology, Zhejiang University, Hangzhou, 310012, China
[b] ZJU-Hangzhou Global Scientific and Technological Innovation Center, Hangzhou, 310012, China
[c] School of Software Engineering, South China University of Technology, Guangzhou, 510006, China
[d] Department of Electrical and Computer Engineering, University of Nevada, Las Vegas, 89154, USA
[e] School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK
[f] School of Electronics and Information Engineering, South China University of Technology, Guangzhou, 510640, China
[g] School of Electronic Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, China

ARTICLE INFO

Keywords:
Countermeasure
Inaudible sound covert channel (ISCC)

ABSTRACT

When an inaudible sound covert channel (ISCC) attack is launched inside a computer system, sensitive data are converted to inaudible sound waves and then transmitted. The receiver at the other end picks up the sound signal, from which the original sensitive data can be recovered. As a forceful countermeasure against the ISCC attack, strong noise can be used to jam the channel and literally shut down any possible sound data transmission. In this paper, enhanced ISCC whose transmission frequency can be dynamically changed is proposed. Essentially, if the transmitter detects that the covert channel is being jammed, the transmitter and receiver both will switch to another available frequency and re-establish their communications, following the proposed communications protocol. Experimental results show that the proposed enhanced ISCC can remain connected even in the presence of a strong jamming noise source. Correspondingly, a detection method based on frequency scanning is proposed to help to combat such an anti-jamming sound channel. With the proposed countermeasure, the bit error rate (BER) of the data communications over enhanced ISCC soars to more than 48%, essentially shutting down the data transmission, and thus neutralizing the security threat.

1. Introduction

Among many serious security vulnerabilities facing industrial and consumer use of near-field communications technologies, covert channels can be particularly hard to be detected and thwarted due to their stealthy nature. Various covert channels can be built by exploring many different types of physical media, including, but not limited to, heat, light, and sounds. The sounds that fall into a special frequency range (>18 kHz) [1] cannot be heard by human ears and a channel thus built is named an inaudible sound covert channel (ISCC). There have been a number of successful implementations of ISCCs built by commercially available hardware [2]. For example,

* Corresponding author.
E-mail addresses: xiaohangwang@zju.edu.cn (X.-H. Wang), li_shoubin@qq.com (S.-B. Li), yingtao.jiang@unlv.edu (Y.-T. Jiang), a.k.singh@essex.ac.uk (A.K. Singh), eebyma@scut.edu.cn (B.-Y. Ma), huanglt@uestc.edu.cn (L.-T. Huang), mei.yang@unlv.edu (M. Yang), csguofen@scut.edu.cn, mei.yang@unlv.edu (F. Guo).

https://doi.org/10.1016/j.jnlest.2022.100181
Received 27 July 2021; Received in revised form 30 November 2022; Accepted 7 December 2022
Available online 9 December 2022

an inaudible covert channel can be established by employing the microphone (available in literally every laptop computer and smartphone nowadays) as the receiver and the speaker as the audio transmitter.

One possible way to combat ISCC is through noise jamming, as illustrated in Fig. 1. A jammer, which can be a mobile phone or a computer, continuously monitors a channel occupying a specific communications frequency band. If the jammer finds that the signal exceeds a preset threshold, it can conclude a covert channel attack is occurring. In order to block the communications over the covert channel, it sends strong noise over the same bandwidth as the sound signal. As a result, the receiver experiences a high bit error rate (BER) that the data transmission is severely crippled or even becomes entirely impossible.

Although jamming can effectively block the data transmission in an inaudible covert channel, an anti-jamming technique, referred to as the adaptive sound frequency selection, can be applied to make the channel much more resilient to jamming noise. Fig. 2 (a) shows the process of an anti-jamming covert channel. Data are transmitted by binary frequency shift keying (FSK) that uses bits in 18 kHz and 19 kHz to represent the states of "0" and "1". As a jammer also sends strong noise with a fixed transmission frequency of 18 kHz, BER of the receiver balloons. However, as shown in Fig. 2 (b), if the frequencies of the covert channel transmission tune to 21 kHz for the bit "0" and 22 kHz for the bit "1", the jamming noise, with its energy centered around the frequency band between 18 kHz and 19 kHz, will not be able to corrupt the transmitted data.

In this paper, such an enhanced covert channel with anti-jamming capability is described along with its communications protocols. In particular, an adaptive frequency selection scheme is adopted, where both the transmitter and receiver keep a list of available transmission frequencies, and they keep ploughing the list until they both tune to the same frequency to establish a connection. In the literature, there are a few protocols proposed to help the covert channel recover from being jammed [3,4]. In Ref. [3], a network covert channel used the timing of packets for encoding, and in Ref. [4] the covert channel set up via audio files and the protocol was used to manipulate the data inside the audio files. Different from them, our paper targets a covert channel that uses the sound as the communications medium and transmits data wirelessly.

To combat an enhanced covert channel with anti-jamming capability, we subsequently propose a countermeasure using a frequency scanning-based detection method. First, a detector constantly monitors the frequency domain to check whether there is an attack or not. Once an attack is discovered, the noise with the same frequency is emitted to jam the current transmission channel. With a high scanning speed, the detection and jamming can track down any frequency change, and correspondingly, continue to block any data traffic over the covert channel, as the case shown in Fig. 2 (c).

Experimental results confirm that an enhanced attack is able to achieve BER as low as 10% with a fixed frequency jammer, leaving the attacker with a reasonably good channel for a dangerously sustained attack. However, with the proposed countermeasure, the enhanced attack will experience a very high BER value, up to 48%, which literally shuts down meaningful covert communications.

Although baseline ISCC was already studied in the literature, noise jamming is not considered against this covert channel. In this paper, an enhanced covert channel is proposed to avoid being jammed, which poses a severe threat to computer systems, followed by a countermeasure to combat it.

The remainder of the paper is organized as follows. Section 2 reviews the related work, and Section 3 introduces baseline ISCC. An enhanced sound covert channel with anti-jamming capability is detailed in Section 4. We propose a countermeasure to fight against such an enhanced sound covert channel in Section 5, and Section 6 presents the experimental results. Finally, Section 7 concludes the paper.

2. Related works

Various communications media, including heat [5–7], cache timing [8–12], light [13–16], and infrared light [17], have found their use in forming covert channels for data communications. Compared with a sound covert channel, a heat covert channel runs a frequency far lower than that of a sound channel, resulting in its transmission throughput typically being much lower than that of its sound counterpart. The light-based covert channel is limited in its use since the travel of light signals can be easily blocked by the obstacles between the transmitter and receiver.

Unlike an infrared-light-based channel that needs special devices (the infrared camera or sensor) to receive signals, a sound covert channel can be easily established using microphones which are readily available in most smartphones and laptops. For embedded

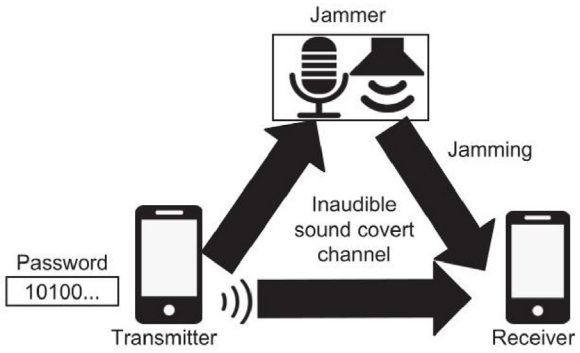


Fig. 1. Illustration of ISCC that is influenced by a jammer.

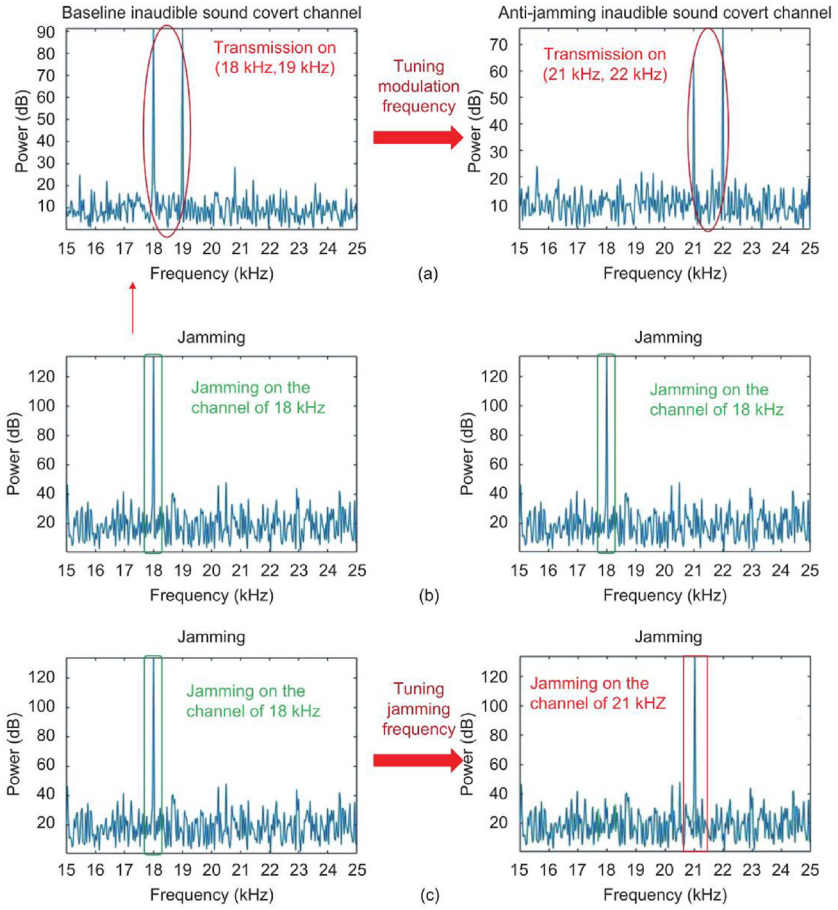


Fig. 2. Illustration of the proposed method: (a) tuning modulation frequency to avoid the channel being jammed, (b) jamming noise, (c) jammer successfully blocks the channel by tracking the frequency of ISCC.

systems, the ISCC attack is found to cause a lot of damages [1], particularly when the covert channel involves inaudible sound. In Ref. [18], it was demonstrated that ISCC is able to hijack smartphones.

2.1. Sound covert channel structure

Acoustic communications over a sound covert channel occur at three layers: The physical layer, data link layer, and application layer.

In the physical layer, various sound players and receivers have been adopted for different purposes. For example, for underwater acoustic communications, transducers and hydrophones are used as the sound player and receiver [19], while in air-gapped systems, speakers and microphones are typically used. In addition to hardware devices, acoustic communications over a sound covert channel can take advantage of many different digital modulation methods [19–21], including on-off keying (OOK), binary frequency shift keying (2FSK), binary phase shift keying (2PSK), and orthogonal frequency division multiplexing (OFDM).

At the data link layer of a sound-based communications system, various communications protocols are available to set up communications sessions, and most of them rely on some form of handshaking.

In the application layer, acoustic communications systems can be used for different purposes and application scenarios. For instance, acoustic signals can enable data to be transmitted underwater [19] or over the air [20,22,23], can be used to build ISCC [21,24–26], and can be used to hijack mobile computing systems [18,27].

Following this three-layer model, we define a baseline sound covert channel that relies on a request-acknowledgment-based communications protocol [1]. The transmitter initiates data transmission by sending a request packet REQ towards the receiver, and the receiver, upon receiving it, replies with an acknowledgment packet ACK to establish a transmission. Once a data transmission session starts, data can be flowing from the transmitter to the receiver until the transmitter sends a terminate packet TER to the receiver to the session.

During the transmission, the transmitted data (baseband) are modulated and emitted as sound signals. The receiver filters out the noise, de-modulates the signal, and finally recovers the original data.

The entire baseline sound covert channel is shown in Fig. 3, and its protocol details are provided in Section 3.

2.2. Anti-jamming using frequency hopping spectrum spreading (FHSS)

Using a jammer, ISCC can be countered by jamming it. A jammer is referred to as a fixed-frequency jammer, as it emits the noise that spans a fixed narrow frequency band. A full-band jammer, on the other hand, emits the jamming noise that spans the entire spectrum, but with very high power consumption.

Jammer-enabled jamming can be an effective means to thwart a sound covert channel. To make the sound covert channel more resilient to jamming, various anti-jamming techniques have been adopted. These techniques have found roots in wireless communications. One such technique is FHSS, which has a long history of being used to avoid jamming or increase bandwidth utilization [2]. In FHSS, the transmitter and receiver change the transmission frequency during the transmission following a specific frequency hopping pattern.

One challenge of the conventional FHSS technologies [28] is their implementation and synchronization costs. To synchronize the FHSS transmitter and the receiver, either a dedicated channel is used to transmit the control message (e.g., the next carrier frequency to be used), or the synchronization signals have to be embedded into the transmission data over the same channel. In fact, conventional FHSS methods can fail the entire transmission, when the channel for transmitting the control messages is jammed. In addition, synchronization in FHSS for anti-jamming sound covert channels often incurs high computation overhead. Such high implementation and synchronization costs in FHSS make it particularly challenging to keep the covert channel stealthy, as the channel prefers to be lightweight and bear a low implementation cost so that little traces are left for the scrutinizer (e.g., a security-aware operating system that is able to check process anomalies) to detect its existence.

3. Baseline ISCC

As shown in Fig. 3, ISCC has two components: The transmitter and receiver. Their functionalities and signaling are described here.

3.1. Transmitter

A transmitter can be a speaker of a laptop or a phone that generates and emits sound signals following a process defined below.

- The transmitter first grabs the data (e.g., password) to be transmitted.
- It encodes the data with errors and packetizes them with added error control codes (ECCs).
- The data packet is modulated by a carrier frequency (typically at a frequency between 18 kHz and 20 kHz). Either amplitude shift keying (ASK) or frequency shift keying (FSK) can be used for modulation, and the latter is used in this paper. In FSK, bit "0" is encoded as a lower frequency sound signal (e.g., 18 kHz), and "1" is encoded as a higher frequency sound signal (e.g., 19 kHz).

The signal to be transmitted, $f(t)$, is thus given as,

$$f(t) = r_1(t)\cos(\omega_1 t) + r_2(t)\cos(\omega_2 t) \tag{1}$$

where

$$r_1(t) = \begin{cases} 1 & \text{sending } ''1'' \\ 0 & \text{sending } ''0'' \end{cases} \tag{2}$$

$$r_2(t) = \begin{cases} 0 & \text{sending } ''1'' \\ 1 & \text{sending } ''0'' \end{cases} \tag{3}$$

and t is time, and ω_1 and ω_2 are angular frequencies.

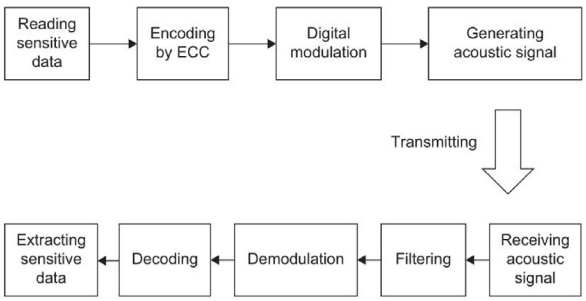


Fig. 3. Baseline sound covert channel.

3.2. Receiver

The receiver picks up the sound signal coming to its receiving device, such as a microphone. The front end of the receiver has two disjoint paths (see Fig. 4), one for the extraction of the high-frequency component h_1 of the signal received, and the other for the low-frequency component h_2. At each path, the signal passes through a band-pass filter (BPF) with the center frequency of h_i and the bandwidth of B_i, followed by a mixer with the oscillation frequency of w_i, where $i = 1$ or 2. After passing the mixer, the signal proceeds to a low-pass filter (LPF). The outputs from the two signal paths converge at the decision-making module, which decides whether the received bit is "1" or "0". Specifically, the receiver works as follows:

1) The sound signal is recorded and saved as a wave file.
2) The band-pass filters (BPF1 and BPF2) are used to filter out the signals according to the carrier frequencies.

The received signal, $s(t)$, is written as

$$s(t) = f(t) + n(t) \tag{4}$$

where $f(t)$ is the actual signal defined in (1), and $n(t)$ is the noise.

The output of BPF1 is given as

$$s_1(t) = r_1(t)\cos(\omega_1 t) + \alpha_1(t) \tag{5}$$

where $\alpha_1(t)$ is the output noise of BPF1.

In the same token, BPF2 output is given as

$$s_2(t) = r_2(t)\cos(\omega_2 t) + \alpha_2(t) \tag{6}$$

where $\alpha_2(t)$ is the output noise of BPF2.

The signal output from the respective mixer is given as,

$$\begin{aligned} & s_i(t)\cos(\omega_i t) = r_i(t)\cos^2(\omega_i t) + \alpha_i(t)\cos(\omega_i t) \\ & = \frac{1}{2}r_i(t) + \frac{1}{2}s_i(t)\cos(2\omega_i t) + \alpha_i(t)\cos(\omega_i t) \quad i = 1, 2. \end{aligned} \tag{7}$$

The output signal of the mixer gets filtered by passing through a low pass filter with a cut-off frequency of h_i ($i = 1, 2$). The output of LPF, $s_{0,i}(t)$, is given as

$$s_{0,i}(t) = \frac{1}{2}r_i(t) + \frac{1}{2}n_{c,i}(t) \; i = 1, 2. \tag{8}$$

where $n_{c,i}(t)$ is the output noise of LPFi ($i = 1, 2$) [29].

The decision maker makes a hard decision by comparing the signal amplitudes from the two signal paths. If the amplitude of the upper path is higher than that of the lower path, the received bit is deemed as bit "1"; otherwise, bit "0".

The transmitter modulates the data into a bitstream as a waveform audio file format (.wav) file. The transmitter then drives its speaker to emit the signal. On the receiver side, it records the audio signal from its microphone and demodulates it to extract the original sensitive data.

3.3. Communications protocol

As shown in Fig. 5, the transmitter sends a REQ packet to initiate a data transfer, and the receiver responds with an ACK packet to establish a communications session. Upon receiving the ACK packet, the transmitter sends its first data packet to the receiver. If the receiver does not receive the data packet, it sends an NACK packet back to the transmitter to request to resend. If the receiver does

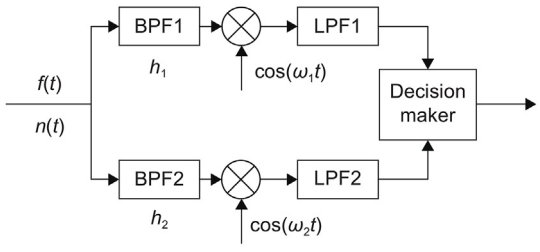


Fig. 4. General receiver structure.

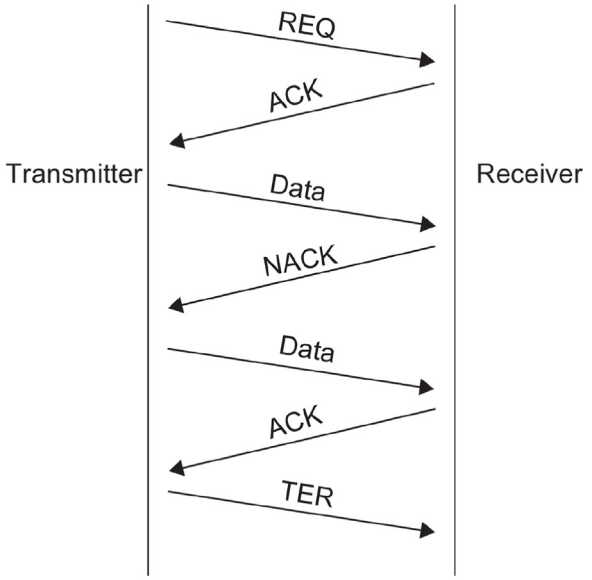


Fig. 5. Message flow of the communications protocol of the baseline channel.

Table 1
Algorithm1.

Protocol to support anti-jamming in the transmitter
Input: F: The set to available frequencies $\{f_0, f_1, \cdots, f_n\}$ Data: The data to be transmitted to the receiver Count: The number of successively received NACKs Buffer_size: The number of bits to be transmitted each time
$i = 0$ // The initial frequency is set to be f_0
Count = 0
NextData = the first Buffer_size bits of Data // first data packet
while Data is not empty **do**
if received ACK (f_i, NULL) **then** // sending the next data packet if ACK is received
NextData = the next Buffer_size bits of Data
send NextData
Count = 0
end if
if NACK (f_i, NULL) is received **then**
if Count > 1 **then** // successively receiving NACK more than once
send AFQ (f_i, f_{i+1})
// select the next available frequency in F if receiving no AFA within time T
while no AFA (f_i, f_j) is received within a given time T **do**
$i = j$ // frequency $j = i + 1$ or randomly selected from F
if $i > n$ **then**
$i = 0$
end if
adjust frequency to f_i
Count = 0
send AFQ (f_i, f_j)
end while
adjust the filter center frequency to be f_j
// selecting f_{i+1} as carrier frequency if receiving AFA
$i = i + 1$
send REQ (f_i, NULL)
else
resend NextData with f_i // resending if receiving NACK once
Count = Count + 1
end if
end if
end while

receive the data, it needs to respond to the sender with an ACK packet. To end the communications session, a TER packet is sent.

4. Anti-jamming sound covert channel

4.1. Communications protocol to support modulation frequency tuning

Assume that the set $F = \{f_0, f_1, \cdots, f_n\}$ contains all the available frequencies for a sound covert channel. The transmission frequencies of the FSK signal f_i fall into the range of (h_1, h_2), with h_1 and h_2 being the highest and lowest available frequencies, respectively.

$$h_i = h_0 + a_i \Delta h \quad i = 1, 2 \tag{9}$$

where h_0 is the lowest frequency in F, and the frequency goes by a_i times of the increments of Δh.

The communications protocol works as follows. Initially, the transmitter initializes the transmission by sending a REQ (f_0, NULL) packet, indicating that the transmitter operates on the frequency f_0 (the first available frequency, also the lowest). The receiver responds by sending an ACK (f_0, NULL) packet to confirm that the following packets can be transmitted on the frequency f_0. Since both the receiver and the transmitter agree to operate on the same frequency, f_0, a communications session is established.

For every packet sent from the transmitter to the receiver, the receiver, upon receiving it, has to send an acknowledgment back to the transmitter. Once the communications are over, the transmitter sends a TER packet to terminate the communications.

During the transmission, the transmitter, after sending a packet, waits for the receiver's ACK or NACK. Upon receiving ACK (f_i, NULL), it starts to transmit the next data packet (as shown in Table 1, Algorithm 1, lines 5 to 9). But if the transmitter receives NACK (f_i, NULL), it resends the last packet (Algorithm 1, lines 27 and 28). If the receiver receives more than one NACK packet, it needs to switch to

Table 2
Algorithm 2.

Protocol to support anti-jamming in the receiver

```
Input: F: The set of available frequencies {f0, f1, ···, fn}
1:    i = 0 // The initial frequency is f0
2:   while TER is not received do
3:        if no REQ (fi, NULL) is received within a given time T then
4:             adjust frequency to fi+1
5:             i = i + 1 // select the next available frequency in F
6:             if i > n then
7:                  i = 0
8:             end if
9:        end if
10:       if REQ (fi, NULL) is received then
11:            send ACK (fi, NULL)
12:       end if
13:       if data packets are received then
14:            if BER > T1 then // send NACK if BER > T1
15:                 send NACK (fi, NULL)
16:            else
17:                 send ACK (fi, NULL)
18:                 extract the data from the data packet
19:            end if
20:       end if
21:          // send AFA and adjust the frequency if receiving AFQ
22:       if AFQ (fi, fj) is received then
23:            send AFA (fi, fj)
24:            select fj as the carrier frequency
25:            i = j
26:       end if
27: end while
```

a different transmission frequency by going through the modulation frequency tuning process (Algorithm 1, lines 11 to 25). And in Table 1, AFQ(f_i, f_j) means the dynamic frequency tuning request and AFA(f_i, f_j) means the dynamic frequency tuning acknowledgment.

On the receiver end, if it finds BER of the received packets exceeds the threshold T_1, it sends NACK (f_i, NULL) (as shown in Table 2, Algorithm 2, lines 14 and 15) towards the transmitter. Once NACK is received by the transmitter, it resends the last packet. If BER of the resent packet is still higher than T_1, the receiver sends another NACK (f_i, NULL). Upon receiving the second NACK, the transmitter starts to change its transmission frequency following a procedure as follows.

Step 1: Assume currently the transmission uses the frequency f_i. If the transmitter needs the receiver to switch to the frequency f_j (frequency $f_j = f_{i+1}$ or randomly selected from F), it sends a dynamic frequency tuning request AFQ (f_i, f_j) to the receiver (Algorithm 1, line 12).
Step 2: After the receiver receives AFQ (f_i, f_j), it adjusts its filter center frequency to be f_j and replies the transmitter with a dynamic frequency tuning acknowledgment AFA (f_i, f_j) (Algorithm 2, lines 22 to 25).
Step 3: After the transmitter receives AFA (f_i, f_j), its carrier frequency switches to f_j (Algorithm 1, lines 23 and 24).
Step 4: The transmitter reinitializes the transmission by sending new REQ (f_j, NULL). The receiver then sends back ACK (f_j, NULL). Now the communications session between the transmitter and receiver is resumed at f_j.

In the worst case, the jamming noise may cause some of the packets incorrectly decoded. To solve this problem, the transmitter and receiver set up a timer T_{tran} and T_{rec} operating at f_i, respectively. The threshold for T_{tran} and T_{rec} is set to T. If the communications are not terminated, but the receiver receives no packets at f_i after T_{rec} (Algorithm 2, lines 3 to 8) expires, it iteratively runs through $f_{i+1}, f_{i+2}, \cdots, f_n$ to see if there is a frequency tuning request.

The transmitter waits until its timer T_{tran} is out, and sends AFQ (f_{i+1}, f_{i+2}) using the frequency f_{i+1}, after which the timer is reset. If the transmitter cannot receive an ACK packet when T_{tran} expires, it continues to send AFQ (f_{i+2}, f_{i+3}) using the frequency f_{i+3}, while resetting T_{tran}. Both the transmitter and receiver keep monitoring or sending requests at the next available frequency until they are synchronized at the frequency f_j to resume their communications.

In this proposed protocol, the following packets are needed, Data, ACK, NACK, REQ, AFQ, and AFA, and their formats are shown in Fig. 6, where:

ACK flag: 1 bit. The flag is set as "1" if it is an ACK/AFA packet, or reset as "0" otherwise.
AF flag: 1 bit. The flag is set as "1" if it is an AFQ/AFA packet, or reset as "0" otherwise.
Data: This is the actual data payload.
AF code: Binary representation of the corresponding frequency.
ECC code: It is used to defend against data corruption.
Preamble: A bit pattern that contains consecutive "1″s and "0″s to indicate the packet begins.

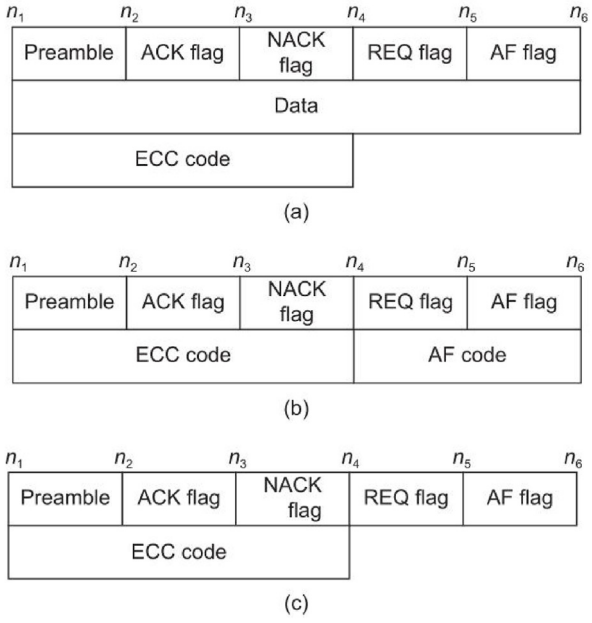


Fig. 6. Packet formats of (a) Data, (b) AFQ and AFA, and (c) ACK, NACK, and REQ.

4.2. Transmitter module

The transmitter operates on the frequency f_i. The frequency tuning process at the transmitter side is triggered, when the transmitter receives AFA (f_i, f_j), or the transmitter receives no acknowledgment packets within T_{tran}.

The transmitter receives AFA (f_i, f_j), after it sends AFQ (f_i, f_{i+1}) to the receiver and adjusts its carrier frequency to f_{i+1}. If it does not receive AFA (f_i, f_j) within T_{tran}, it keeps sending AFQ (f_{i+1}, f_{i+2}), AFQ (f_{i+2}, f_{i+3}), ⋯ continuously until it gets a reply from the receiver.

4.3. Receiver module

The receiver operates on the frequency f_i (see Fig. 7). It needs to tune the center frequency of its filter in two scenarios, as shown in Fig. 8:

1) When the transmitter receives AFQ (f_i, f_j) from the transmitter, the receiver adjusts the center frequency of its filter to be f_{i+1} and replies the transmitter with AFA (f_i, f_j).
2) If the transmitter does not receive REQ (f_j, NULL) after it tunes its frequency within T, the receiver iteratively tunes the center frequency of the filter to f_{i+2}, f_{i+3}, ⋯, until it receives a request packet.

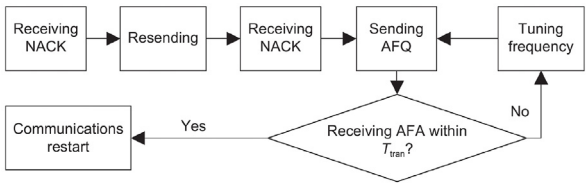


Fig. 7. Transmitter of the anti-jamming covert channel tunes frequency.

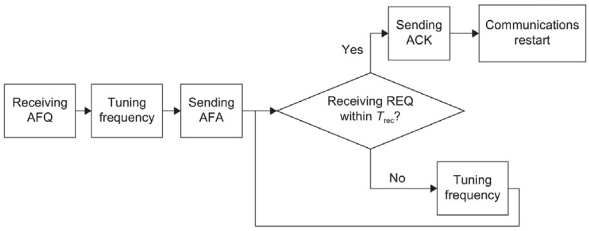


Fig. 8. Receiver of the anti-jamming covert channel tunes frequency.

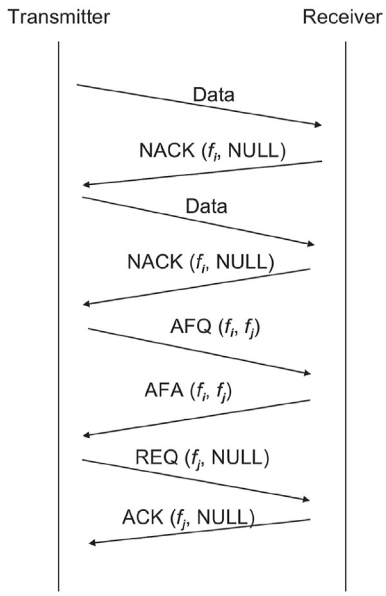


Fig. 9. Message flow of an anti-jamming covert channel.

The data flow of anti-jamming ISCC is shown in Fig. 9. Assume the covert channel is using f_i to transmit packets and the receiver detects that jamming is in effect. It sends NACK (f_i, NULL) to the transmitter, which demands the transmitter to resend the data packet previously sent. If the receiver detects the jamming is persistent, it will send NACK (f_i, NULL) to the transmitter again. The transmitter then sends AFQ (f_i, f_j) to the receiver to initiate another round of frequency tuning. The receiver replies by sending AFA (f_i, f_j) toward the transmitter, and both need to switch to f_j. Then, the transmitter sends REQ (f_j, NULL) and the receiver responds with ACK (f_j, NULL), which resumes the communications.

The data and control packets can work at two separate channels, such that when the data channel is jammed, the control packets can still be delivered. Both channels can tune their operation frequencies.

5. Countermeasure

As the intuitive countermeasure in response to an enhanced sound covert channel described in the previous section, full-band jamming, whose bandwidth of the jamming noise covers the entire frequency band F of the audio covert channel, can be applied. However, this indiscriminate approach comes with excessive power consumption. Instead, a lightweight countermeasure, which is much more power efficient, is proposed.

As shown in Fig. 10, the proposed countermeasure scheme involves a detector and a jammer. The detector is able to let BPF's center frequency f_d be tuned between J_1 and J_2. In practice, 18 kHz $\leq J_1 < J_2 \leq$ 22 kHz. Once a possible covert channel signal centered around f_d is detected, the jamming noise with f_d is emitted to jam the channel and block any data transmission.

5.1. Detector

The detector performs the following steps.

Step 1. Scanning in the frequency domain

The detector continuously picks up the audio signals using an acoustic sensor, such as a microphone. BPF with the center frequency f_i and the bandwidth of 1 kHz is used to filter out the audio signals outside the spectrum (as shown in Table 3: Algorithm 3, line 3). The center frequency f_d of BPF increases in a stepwise manner, that is, $f_i(n) = f_i(n-1) + \Delta f$ (Algorithm 3, line 7), where $f_d(n)$ and $f_d(n-1)$ are the center frequencies at the nth and $(n-1)$th steps of the scanning. The scanning will cycle through the entire frequency range of [J_1, J_2].

Step 2. Decision making

If the amplitude of the output signal from BPF exceeds the threshold T_d (Algorithm 3, lines 4 to 5), it is deemed that a covert channel is present. If a covert channel is detected, the jammer emits the noise spanning the same frequency used by the covert channel.

5.2. Jammer

Once the sound covert channel is detected, the jammer starts the jamming process, which has the following steps.

Step 1. Random noise generation

Once ISCC is detected, a pulse train (i.e., a sequence of consecutive "1"s) is generated to jam the covert channel. This sequence $s_j(t)$ can be expressed as

$$s_j(t) = g_j\left(t - nT_j\right) \tag{10}$$

where T_j is the period of one bit (symbol width), n is the random natural number, and $g_j(t)$ is the baseband pulse waveform with a

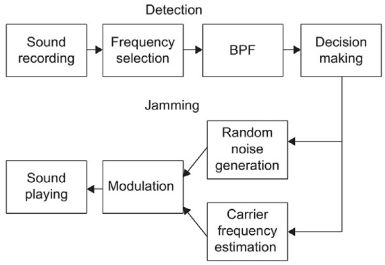


Fig. 10. Structure of the jammer.

Table 3
Algorithm 3.

The scanning approach of the jammer

Input: F: The set of available frequencies $\{f_0, f_1, \cdots, f_n\}$
Δf: The frequency increment of linear scanning
f_d: The center frequency of BPF
Output: f_j: The detected sound signal frequency
$f_d = f_0$
while $f_d \leq f_n$ **do**
Filter the sound signal using BPF with f_d, and the bandwidth of Δf
if the maximum of the filtered signal amplitude $\geq T_d$ **then**
$f_j = f_d$
else
$f_j = f_d + \Delta f$
end if
end while
return f_j

duration of T_j.

Step 2. Modulation

The noise sequence $s_j(t)$ is modulated for transmission as the sound signal with the carrier frequency f_j. The sound signal $e_j(t)$ is thus

$$e_j(t) = s_j(t)c(t) \tag{11}$$

where $c(t)$ is the carrier signal.

Step 3. Sound playing

The noise is played by the jammer's audio device (e.g., a speaker). The noise at the frequency f_j thus causes the receiver's decoder to fail to decode the signal received.

6. Experiment and evaluation

6.1. Experiment setup

Three laptop computers are used to run the experiments, one serving as the transmitter, one as the receiver, and the third as the jammer. All three have the same configurations, as tabulated in Table 4. The laptop model is Hasee Z6-KP7S1. In our experiments, we compare the transmission BER and throughput measured in bit per second (bps). Table 5 summarizes all the parameters as described in Sections 3, 4, and 5.

In the experiments, the transmitter and receiver operate with four frequencies, and their respective codes are shown in Table 6, which are the AF code field of the data packets.

The transmitter sets up its timer T_{tran}, and the receiver sets up its timer T_{rec}. The threshold for both T_{tran} and T_{rec} is set to be T. If T is too short, the covert channel adjusts the frequency frequently, which reduces the throughput. If T is too long, when the covert channel is seriously jammed, it cannot adjust the transmission frequency quickly enough, which can adversely impact the transmission throughput. Fig. 11 shows how the throughput varies with the value of T. From Fig. 11, it can be seen that the throughput of the anti-jamming sound covert channel reaches its highest performance with $T = 2$ s. So, in the following experiment, we set T_{tran} and T_{rec} to be 2 s.

Table 4
Configuration of the three laptops in experiments.

Items	Parameter
Processor	Intel® Core™ i7-7700HQ CPU @ 2.80 GHz
Sound card	Realtek high definition audio
Operating system	64-bit Windows
RAM size	8 GB

Table 5
Detailed parameters.

Parameter	Value	Parameter	Value
B_i	h_i/100	T_d	50%
h_0	18 kHz	Δh	1 kHz
a_i	0, 1, 2, 3	T	2 s
T_1	20%	T_j	0.7 s
n_1	0	n_2	4
n_3	5	n_4	6
n_5	7	n_6	8
F	(18 kHz, 19 kHz), (18 kHz, 20 kHz), (18 kHz, 21 kHz), (19 kHz, 20 kHz), (19 kHz, 21 kHz), (20 kHz, 21 kHz)	/	/

Table 6
A 2-bit code represents a frequency in a data packet.

Code	Frequency
00	18 kHz
01	19 kHz
10	20 kHz
11	21 kHz

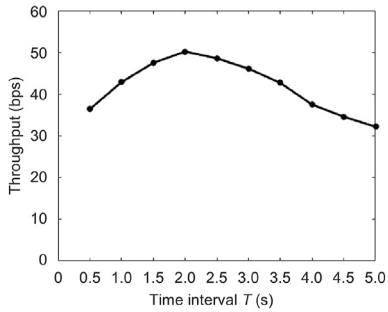


Fig. 11. Throughput vs. T.

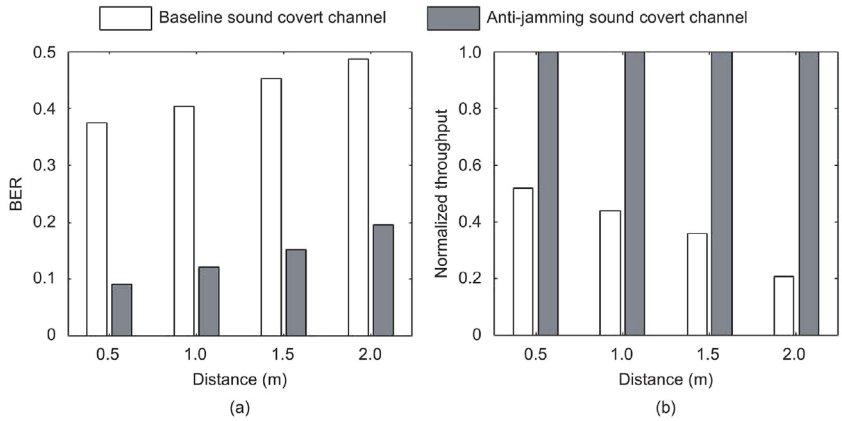


Fig. 12. BER and throughput comparison when the transmitter and receiver are at different distances: (a) BER and (b) throughput (normalized).

6.2. Evaluating anti-jamming covert channel

6.2.1. Evaluating with different distances

In this set of experiments, the distance between the transmitter and receiver varies from 0.5 m to 1 m, 1.5 m, and 2 m. The jammer works at a frequency randomly chosen from the band given in [18 kHz, 22 kHz]. The throughput thus obtained is normalized with respect to that of the proposed anti-jamming covert channel.

Fig. 12 compares the proposed anti-jamming covert channel against the baseline covert channel [1] by varying the communications distance with a fixed jammer working at a frequency randomly chosen from the band that falls into [18 kHz, 22 kHz]. As shown in Fig. 12 (a), with a distance of 0.5 m, the proposed anti-jamming covert channel can reduce BER by 72.9% over the baseline covert channel. The proposed anti-jamming covert channel shows its BER 67.2% lower than the baseline covert channel, as the distance stretches to 2 m. The drop of BER with the increase of the transmission distance is due to the decay of the sound signal as the transmitter and receiver are farther apart. As the proposed covert channel can change its transmission frequency, it is less impacted by jamming, and its BER is quite low, largely regardless of the transmission distance.

As shown in Fig. 12 (b), the throughput of the anti-jamming covert channel is higher than that of the baseline covert channel. At a long distance, in this case, the transmitter and receiver are 2 m apart, which shows that the throughput of the proposed covert channel is 5 times of that of the baseline covert channel.

6.2.2. *Evaluating with different signal noise ratios (SNRs)*

In this set of experiments, SNR varies from 10 dB, 20 dB, 30 dB, 40 dB, 50 dB, 60 dB, to 70 dB by changing the jamming noise power. The jammer works at a frequency randomly chosen from the band of [18 kHz, 22 kHz]. The throughput is normalized with respect to that of the anti-jamming covert channel.

In Fig. 13 (a), with SNR at 10 dB, the proposed anti-jamming covert channel can reduce BER by 63.2% over the baseline covert channel [1]. The proposed anti-jamming covert channel can reduce BER by 77.1% over the baseline covert channel, as SNR increases to 70 dB. With higher SNR, the noise impact becomes less significant. In sharp contrast, the baseline covert channel without the anti-jamming capability can be easily compromised by the jammer with its strong jamming noise. Since the proposed covert channel can change its transmission frequency when it senses the jamming, it immediately dodges the jamming noise to deliver a high throughput performance.

As shown in Fig. 13 (b), the throughput of the anti-jamming covert channel is higher than that of the baseline covert channel. With 70-dB SNR, the throughput of the proposed covert channel is 4 times of that of the baseline covert channel. This is because the proposed anti-jamming covert channel is able to reduce transmission BER, which can be translated into the suppression of time, otherwise data retransmission is needed.

6.2.3. *Comparison with the existing audio covert channels*

In this set of experiments, we compare the proposed anti-jamming covert channel with four existing audio covert channels, the baseline sound covert channel [1], the covert acoustical mesh network [20] which is established by multi-hop communications between different sound covert channel participants, and two other channels that incorporate FHSS. The latter two channels are referred to as FHSS-A (a channel that uses an independent channel at 18 kHz for synchronization) [28], and FHSS-B (conventional FHSS using the same communications channel of data transmission for synchronization) [2], respectively. These schemes are compared when they are exposed to fixed-frequency jamming noise with different noise power levels. Here the normalized jamming noise power is defined as the ratio of the jammer's noise power to the audio covert channel's communications signal power.

As shown in Fig. 14, BER of FHSS-A is much higher than that of the proposed sound covert channel. The reason is that, once the channel for synchronization is jammed, the whole communications will be shut down completely. Since FHSS-B uses the same communications channel for synchronization, when the frequency of jamming noise matches the communications channel's frequency, both the transmission and synchronization will be blocked by the fixed frequency jamming, making BER of FHSS-B significantly higher than that of the proposed audio covert channel.

When the power of the jamming noise equals that of the sound covert channel signal, BER of the baseline sound covert channel is as high as 49%, while BER of the proposed audio covert channel is only about 10%. The reason is that the jammer can detect the frequency of the baseline sound covert channel and inject the jamming noise in the same frequency, which drags the receiver of the baseline sound covert channel to make incorrect decoding decisions. BER of the covert network in Ref. [19] is lower than the baseline sound covert channel, but much higher than that of the proposed sound covert channel, especially when the power of jamming noise equals that of the sound covert channel signal, BER of the covert network in Ref. [19] is as high as 46%, while BER of the proposed scheme is only 10%. Because when the jamming noise is strong, the transmission in the covert network in Ref. [19] is completely blocked. In sharp contrast, the proposed anti-jamming covert channel can change the frequency of communications dynamically and can finally synchronize with the proposed communications protocol even under strong jamming noise, so its BER being the lowest among all the channels.

6.3. *Evaluating proposed countermeasure*

6.3.1. *Evaluating the detection unit*

In this set of experiments, the scanning speed of a jammer (given as the number of center frequency changes per second), which varies from 1 time/s to 10 times/s, is examined to determine how it is related to detection efficiency against the proposed enhanced ISCC.

From Fig. 15 (a), it can be seen that, as the scanning speed increases, BER of the proposed anti-jamming covert channel rapidly increases. With the scanning speed of the jammer at 1 time/s, BER is only 9.5%; while at the scanning speed of 10 times/s, BER jumps to 47.3%. Because with the increase of the jammer scanning speed, there is an increased probability that the jammer is able to find and thus jams the transmission. As the scanning speed reaches 10 times/s, the detector is able to track the covert channel in a more fine-grained manner, and correspondingly, BER skyrockets.

Also shown in Fig. 15 (b), as the scanning speed gets faster, the throughput of the proposed anti-jamming covert channel decreases.

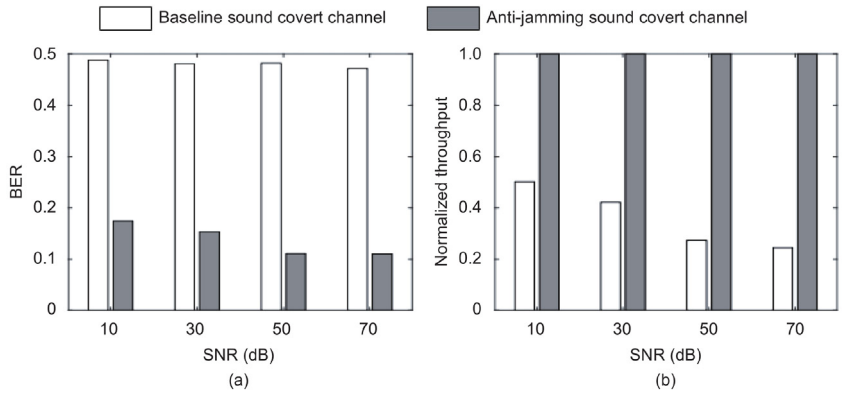


Fig. 13. BER and throughput comparison with different SNRs: (a) BER and (b) throughput (normalized).

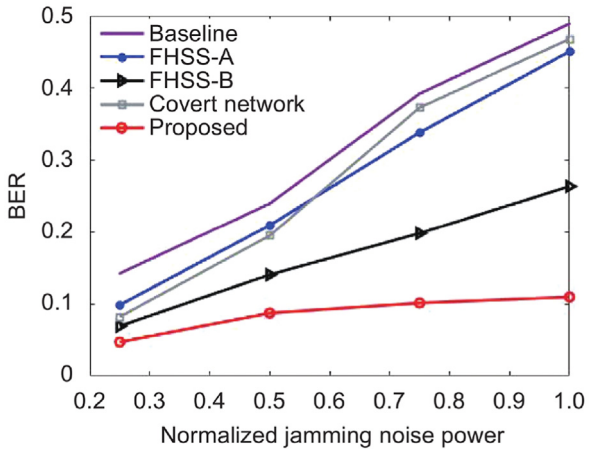


Fig. 14. BER comparison by varying the power of jamming noise.

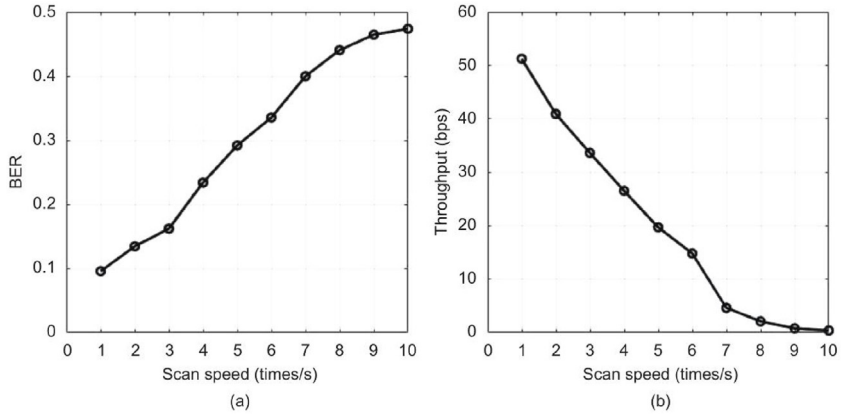


Fig. 15. BER and throughput under different scanning speeds of the jammer: (a) BER (b) throughput.

When the scanning speed is 1 time/s, the effective throughput (valid bits received by the receiver per second) of the proposed covert channel is about 50 bps. In sharp contrast, as the scanning speed increases to 10 times/s, the throughput of the proposed covert channel drops to nearly 0. In this case, the covert channel continues to change its frequency all the time, and therefore, there will be no meaningful data transfers.

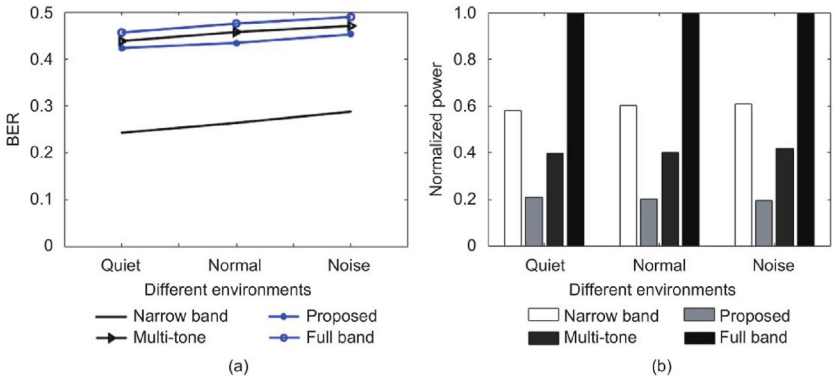


Fig. 16. BER and the power consumption of the anti-jamming covert channel: (a) BER under the four countermeasures and (b) power consumption of the four countermeasures (normalized).

6.3.2. Evaluating countermeasure

In this set of experiments, we compare the following four countermeasures against the proposed anti-jamming ISCC:

1) The first one is a fixed-frequency "narrow-band" jamming with a center frequency of 19 kHz and a bandwidth of 3 kHz.
2) The second one adopts the proposed countermeasure, which can track the communications frequency of ISCC and send jamming noise as detailed in Section 5, with a bandwidth of 1 kHz.
3) The third one is "multi-tone" jamming together with the proposed detection approach. Multi-tone jamming tracks not only the transmission frequency, but also the receiving frequency. Both frequencies of the FSK signal are detected, and the noise at these two frequencies is emitted, with a bandwidth of 1 kHz.
4) The last one is "full-band" jamming, whose bandwidth stretches to the entire communications spectrum, from 18 kHz to 22 kHz.

The experiments are performed in quiet (e.g., a room without background noise and people), normal (e.g., a research lab), and noisy environments (e.g., the public area of a university campus).

Fig. 16 shows BER and the power consumption of the anti-jamming covert channel when the four countermeasures are separately applied. For each of the four countermeasures, its power is determined as the integral of the power spectrum density across the entire frequency band (from 18 kHz to 22 kHz) of the jamming noise, and the result is normalized with respect to that of the full-band countermeasure. It can be seen that, from Fig. 16, for each of the countermeasures, the jamming effects in different environments are similar. This is because the covert channel runs at a frequency (over 18 kHz) that is typically higher than the environmental noise, making the covert channel less impacted by the environmental noise. The jamming effect of the fixed-frequency narrow-band countermeasure is weaker than the others, which leads to maximal BER of only about 25%. The full-band jammer can cover the whole frequency spectrum, which causes BER to reach 50%, but at a cost of high power consumption which is 5 times over that of the proposed scheme on average. On the other hand, the proposed countermeasure can achieve BER of 45% with much lower power consumption than the full-band jamming countermeasure. The multi-tone jamming can also achieve BER of 47%, which is slightly better than the proposed countermeasure, but at a high cost, i.e., its power consumption is twice as much as that of the proposed countermeasure. As shown in Sections 3 and 5, the dual frequency FSK system uses two frequencies to carry bits "0" and "1", and the receiver makes its decision based on the amplitudes of the signals that come with two frequencies. So even if the proposed countermeasure is a single-tone jamming approach, the receiver's decision maker cannot give the right decision, i.e., the proposed jamming makes the receiver always get an all-"1" bitstream. In summary, the proposed countermeasure can effectively defend against the anti-jamming covert channel with a very low cost in terms of power consumption.

7. Conclusions

In this paper, an anti-jamming enhancement for ISCC is proposed to prevent the channel from being jammed. Essentially, an enhanced covert channel relies on its transmission frequency to be dynamically changed, when a jamming noise source is found. Such enhancement will contribute to the BER reduction of the channel, which effectively increases the transmission throughput and makes the channel more robust and thus more dangerous. To combat this enhanced covert channel, a scanning-based detection and a countermeasure are proposed. This countermeasure scheme tracks the transmission frequencies of the covert channel and accordingly emits noise occupying the same frequency band to block any data transmission. Experimental results showed that with a jammer running at a scanning speed of 10 times/s, the throughput of the proposed anti-jamming covert channel drops dramatically, as its BER rises to a remarkably high level, at 48%, which effectively shuts down any meaningful data communications.

Funding

This work was supported partly by the National Natural Science Foundation of China under Grant No. 61971200; partly by Zhejiang Lab under Grants No. 2021LE0AB01 and No. 2021PC0AC01; partly by the Major Scientific Research Project of Zhejiang Lab under Grant No. 2021LE0AC01; partly by the Key Technologies R&D Program of Jiangsu (Prospective and Key Technologies for Industry) under Grant No. BE2021003; partly by the National Key Research and Development Program of China under Grant No. 2019QY0705; supported by the Guangdong Provincial Key Laboratory of Short-Range Wireless Detection and Communication under Grants No. 2014B030301010 and No. 2017B030314003; partly by the National Key R&D Program of China, Grant No. 2022YFB4401403.

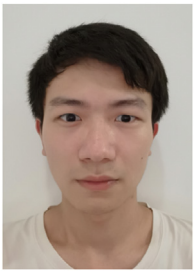
Declaration of competing interest

The authors declare no conflicts of interest.

References

[1] M. Guri, Y. Solewicz, Y. Elovici, MOSQUITO: covert ultrasonic transmissions between two air-gapped computers using speaker-to-speaker communication, in: Proc. of IEEE Conf. on Dependable and Secure Computing, 2018, pp. 1–8. Kaohsiung.
[2] B. Carrara, Air-gap Covert Channels, Ph.D. dissertation, Univ. Ottawa, Ottawa, 2016.
[3] S. Wendzel, The problem of traffic normalization within a covert channel's network environment learning phase, in: Proc. of Sicherheit-Sicherheit, Schutz und Zuverlässigkeit, Bonn, 2012, pp. 149–161.
[4] M. Naumann, S. Wendzel, W. Mazurczyk, J. Keller, Micro protocol engineering for unstructured carriers: on the embedding of steganographic control protocols into audio transmissions, Secur. Commun. Network. 9 (15) (2016) 2972–2985, Oct.
[5] J.-C. Wang, X.-H. Wang, Y.-T. Jiang, A.K. Singh, L.-T. Huang, M. Yang, Combating enhanced thermal covert channel in multi-/many-core systems with channel-aware jamming, IEEE Trans. Comput. Aided Des. Integrated Circ. Syst. 39 (11) (2020) 3276–3287, Nov.
[6] H.-L. Huang, X.-H. Wang, Y.-T. Jiang, A.K. Singh, M. Yang, L.-T. Huang, On countermeasures against the thermal covert channel attacks targeting many-core systems, in: Proc. of the 57th ACM/EDAC/IEEE Design Automation Conf vol. 194, 2020, pp. 1–6. San Francisco.
[7] J. Kwon, S. Kim, S. Roh, B. Lee, Tunable dispersion slope compensator using a chirped fiber Bragg grating tuned by a fan-shaped thin metallic heat channel, IEEE Photon. Technol. Lett. 18 (1) (Jan. 2006) 118–120.
[8] C. Maurice, C. Neumann, O. Heen, A. Francillon, C5: cross-cores cache covert channel, in: Proc. of the 12th Intl. Conf. on Detection of Intrusions and Malware, and Vulnerability Assessment, 2015, pp. 46–64. Milan.
[9] B.B. Brumley, R.M. Hakala, Cache-timing template attacks, in: Proc. of the 15th Intl. Conf. on the Theory and Application of Cryptology and Information Security, 2009, pp. 667–684. Tokyo.
[10] D. Jayasinghe, J. Fernando, R. Herath, R. Ragel, Remote cache timing attack on advanced encryption standard and countermeasures, in: Proc. of the 5th Intl. Conf. on Information and Automation for Sustainability, 2010, pp. 177–182. Colombo.
[11] B.A. Braun, S. Jana, D. Boneh, Robust and efficient elimination of cache and timing side channels [Online]. Available: https://arxiv.org/abs/1506.00189, August 2015.
[12] U. Herath, J. Alawatugoda, R. Ragel, Software implementation level countermeasures against the cache timing attack on advanced encryption standard, in: Proc. of IEEE the 8th Intl. Conf. on Industrial and Information Systems, 2013, pp. 75–80. Peradeniya.
[13] N. Chi, Models of the visible light channel, in: N. Chi (Ed.), LED-based Visible Light Communications, Springer, Berlin, 2018, pp. 39–58.
[14] Y.-F. Yang, Z.-Q. Cao, Q.-S. Shen, et al., Multi-channel light modulation based on the attenuation total reflection, Opt Laser. Technol. 37 (3) (Apr. 2005) 225–228.
[15] W. Liu, X. Zhou, J.-H. Huo, K.-L. Yan, Modeling of visible light channel based on matrix reconstruction, Beijing, in: Proc. of the 4th Intl. Conf. on Wireless and Optical Communications, 2016, pp. 1–5, 990205.
[16] V.N. Gorbachev, A.I. Trubilko, Quantum channel for states of light as based on integrals of motion, J. Exp. Theor. Phys. Lett. 83 (4) (Oct. 2006) 179–184.
[17] A.C.M. van Rijn, A. Peper, C.A. Grimbergen, A wireless infrared link for a 16-channel EEG telemetry system, in: Proc. of the 16th Annual Intl. Conf. of the IEEE Engineering in Medicine, and Biology Society, Baltimore, 1994, pp. 906–907.
[18] L. Deshotels, Inaudible sound as a covert channel in mobile devices, in: Proc. of the 8th USENIX Conf, . on Offensive Technologies, San Diego, 2014, pp. 16–25.
[19] G. Leus, P.A. van Walree, Multiband OFDM for covert acoustic communications, IEEE J. Sel. Area. Commun. 26 (9) (2008) 1662–1673, Dec.
[20] L. Chuan, D.A. Hutchins, R.J. Green, Short-range ultrasonic digital communications in air, IEEE Trans. Ultrason. Ferroelectrics Freq. Control 55 (4) (Apr. 2008) 908–918.
[21] M. Hanspach, M. Goetz, On covert acoustical mesh networks in air, J. Commun. 8 (11) (Nov. 2013) 758–767.
[22] M. Guri, Y. Solewicz, Y. Elovici, Speaker-to-speaker covert ultrasonic communication, J. Inf. Secur. Appl. 51 (Apr. 2020) 1–10, 102458.
[23] S. Soderi, Acoustic-based security: a key enabling technology for wireless sensor networks, Int. J. Wireless Information Networks 27 (1) (Mar. 2020) 45–59.
[24] M. Guri, Cd-Leak, Leaking secrets from audioless air-gapped computers using covert acoustic signals from CD/DVD drives, in: Proc. of IEEE the 44th Annual Computers, Software, and Applications Conf., Madrid, 2020, pp. 808–816.
[25] J.E. Coyac-Torres, M.E. Rivero-Angeles, E. Aguirre-Anaya, Cognitive radio based system for best effort communications in sound-based covert channel for IoT environments, Mobile Network. Appl. 26 (4) (2021) 1449–1460, Aug.
[26] M. Guri, Y. Solewicz, Y. Elovici, Fansmitter: acoustic data exfiltration from air-gapped computers via fans noise, Comput. Secur. 91 (Apr. 2020) 1–14, 101721.
[27] H.-L. Li, Y.-Y. Zou, W.-J. Yi, Z.-Y. Ye, Y. Ma, A covert information transmission scheme based on high frequency acoustic wave channel, in: Proc. of the 6th Intl. Conf. on Artificial Intelligence and Security, Hohhot, 2020, pp. 203–214.
[28] A. Alsadi, S. Mohan, A new frequency hopping scheme to secure the physical layer in the Internet of things (IoT), in: Proc. of Wireless Telecommunications Symposium, 2020, pp. 1–8. Washington.
[29] J.G. Proakis, M. Salehi, G. Bauch, Contemporary Communication Systems Using MATLAB, third ed., Cengage Learning, Stamford, 2012, pp. 313–361.

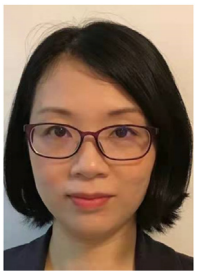
Xiao-Hang Wang received the B.Eng. and Ph.D. degrees in communications and electronic engineering from Zhejiang University, Hangzhou in 2006 and 2011, respectively. He is currently a professor with Zhejiang University. He was the recipient of the Best Paper Awards from *the 23rd Euromicro Intl. Conf. on Parallel, Distributed and Network-Based Processing (PDP)* in 2015 and *the 22nd Intl. Conf. on Very Large Scale Integration (VLSI-SoC)* in 2014. His research interests include the many-core architectures, power-efficient architectures, optimal control, and network-on-chip (NoC)-based systems.



Shou-Bin Li received the B.Eng. degree in software engineering from South China University of Technology, Guangzhou in 2019. He is currently pursuing his M.S. degree with South China University of Technology. His research interests include computer architecture and data security.

Ying-Tao Jiang joined the Department of Electrical and Computer Engineering, University of Nevada, Las Vegas in August 2001, upon obtaining his Ph.D. degree in computer science from The University of Texas at Dallas, Richardson. He has been a full professor since July 2013 with University of Nevada and served as the Department Chair from 2015 to 2018. Since July 2018, he has begun serving as the Associate Dean of the College of Engineering, The University of Texas at Dallas. His research interests include algorithms, computer architecture, VLSI, networking, and nanotechnologies.

Amit Kumar Singh received the B.Tech. degree in electronics engineering from Indian Institute of Technology (Indian School of Mines), Dhanbad in 2006, and the Ph.D. degree from the School of Computer Engineering, Nanyang Technological University (NTU), Singapore in 2013. He was with HCL Technologies (India) for a year and a half before pursuing his Ph.D. degree with NTU in 2008. He worked as a post-doctoral researcher with National University of Singapore (NUS), Singarpore from 2012 to 2014 and with University of York, Heslington from 2014 to 2016. Currently, he is working as an associate professor with University of Essex, Colchester. His current research interests include system level design-time and run-time optimizations of 2-dimensional and 3-dimensional multi-core systems with focuses on performance, energy, temperature, and reliability. He has published over 45 papers in the above areas in leading international journals/conferences.



Bi-Yun Ma received the Ph.D. degree from University of Nantes, Nantes in 2010. She was a visiting professor with Ecole Polytech Nantes, Nantes in 2011. She is currently an associated professor with the School of Electronic and Information Engineering, South China University of Technology. Her main research interests include ultrasonic communications, ultrasonic detection, as well as intra-body networks.

Le-Tian Huang received the M.S. and Ph.D. degrees in communications and information system from University of Electronic Science and Technology of China (UESTC), Chengdu in 2009 and 2016, respectively. He is an associate professor with UESTC. His scientific work contains more than 40 publications including book chapters, journal articles, and conference papers. His research interests include heterogeneous multi-core system-on-chips, network-on-chips, and mixed signal integrated circuit design.

Mei Yang received her Ph.D. degree in computer science from The University of Texas at Dallas in August 2003. She has been a full professor with the Department of Electrical and Computer Engineering, University of Nevada since 2016. Her research interests include computer architecture, networking, and embedded systems.

Fen Guo received the Ph.D. degree in computer application technology from the School of Software Engineering, South China University of Technology in 2015. She has been working as a teacher with South China University of Technology since 2005. Her current research interests include data mining, data security, and cloud computing.