SPEAKE(a)R: Turn Speakers to Microphones for Fun and Profit

Mordechai Guri, Yosef Solewicz, Andrey Daidakulov, Yuval Elovici
Ben-Gurion University of the Negev
Cyber Security Research Center
gurim@post.bgu.ac.il; yosef.solewicz@gmail.com; daidakul@post.bgu.ac.il; elovici@post.bgu.ac.il

Demo video: https://www.youtube.com/watch?v=ez3o8aIZCDM

Abstract

It's possible to manipulate the headphones, earphones, and simple earbuds connected to a computer, silently turning them into a pair of eavesdropping microphones. This paper focuses on the cyber security threat this behavior poses. We introduce 'SPEAKE(a)R,' a new type of espionage malware that can covertly turn the headphones, earphones, or simple earbuds connected to a PC into microphones when a standard microphone is not present, muted, taped,[1] or turned off. We provide technical background at the hardware and OS levels, and explain why most of the motherboards and audio chipsets of today's PCs are susceptible to this type of attack. We implemented a malware prototype and tested the signal quality. We also performed a series of speech and recording quality measurements and discuss defensive countermeasures. Our results show that by using SPEAKE(a)R, attackers can record human speech of intelligible quality and eavesdrop from nine meters away.

1. Introduction

Audio playing equipment such as loudspeakers, headphones, and earphones are widely used in PCs, laptops, smartphones, media entertainment systems, and more. In this section we refer to any audio playing equipment that contains speakers (loudspeakers, headphones, earphones, etc.) as speakers.

Speakers aim at amplifying audio streams out, but a speaker can actually be viewed as a microphone working in reverse mode: loudspeakers convert electric signals into a sound waveform, while microphones transform sounds into electric signals. Speakers use the changing magnetic field induced by electric signals to move a diaphragm in order to produce sounds. Similarly, in microphone devices, a small diaphragm moves through a magnetic field according to a sound's air pressure, inducing a corresponding electric signal [1]. This bidirectional mechanism facilitates the use of simple headphones as a feasible microphone, simply by plugging them into the PC microphone jack. It should be clear that in practice, speakers were *not* designed to perform as microphones and the recorded signals will be of low quality.

1.1. Jack retasking

A typical computer chassis contains a number of audio jacks, either on the front panel, rear panel, or both. These jacks are the sockets for plugging in various audio equipment such as speakers, headphones, and microphones. Each jack is used either for input (line in), or output (line out). The audio ports usually have a conventional coloring system; typically green is used for speakers (output jack), blue for line in (input jack), and pink for microphones (input jack).

Interestingly, the audio chipsets in modern motherboards and sound cards include an option to change the function of an audio port at software level, a type of audio port programming sometimes referred to as jack retasking or jack remapping. This option is available on Realtek's (Realtek Semiconductor Corp.) audio chipsets, which are integrated into a wide range of PC motherboards today. Jack retasking, although documented in applicable technical specifications, is not well-known, as was mentioned by the Linux audio developer, David Henningsson, in his blog:

"Most of today's built-in sound cards are to some degree retaskable, which means that they can be used for more than one thing. …the kernel exposes an interface that makes it possible to retask your jacks, but almost no one seems to use it, or even know about it" [2].

1.2. Microphone-less eavesdropping

The fact that headphones, earphones, and earbuds are physically built like microphones, coupled with the fact

[1] "Why has Mark Zuckerberg taped over the webcam and microphone on his MacBook?" [4]

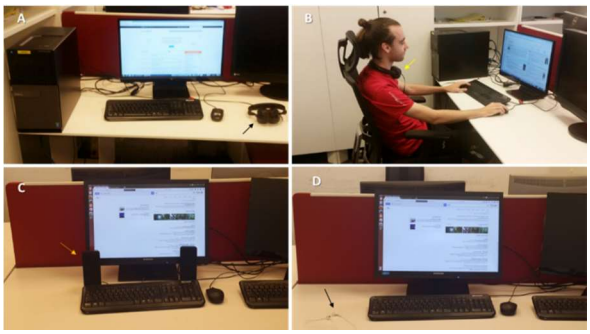


Figure 1. Illustration of SPEAKE(a)R. Headphones (scenarios A, B), speakers (scenario C), and earphones (scenario D) are connected to a computer which has no microphone. A malware running on the computer turns them into microphones. Note that in scenario C the method requires 'passive' loudspeakers, which are rarely in use today.

that an audio port's role in the PC can be altered programmatically, changing it from output to input, creates a vulnerability which can be abused by attackers. A malware can stealthily reconfigure the headphone jack from a line out jack to a microphone jack. As a result, the connected headphones can function as a pair of recording microphones, thereby rendering the computer into an eavesdropping device – even when the computer doesn't have a connected microphone (Figure 1).

In this paper we provide a technical overview of SPEAKE(a)R – a malware that can covertly transform headphones into a pair of microphones – and show how it can be exploited in different scenarios. We also evaluate the malware's efficacy and the recording quality, and present defensive countermeasures.

1.3. Scope of the attack

The attack described in this paper is mainly relevant to headphones, earphones, and simple earbuds. It is also relevant to a type of speaker that is passive (see Section 2.1), without an internal amplifier. Although such speakers are in use today **[3]**, they are much less common in modern desktop PCs. This effectively limits the scope of the attack to headphones, earphones, and earbuds.

1.4. Attack scenarios

In the context of cyber security, the attack scenario is relevant to spyware or malware with an espionage intent. Such spying malware may have the capability of keylogging, stealing files and passwords, and audio recording. There are two main scenarios for using headphones as a microphone.

Microphone-less computers. This scenario involves a PC that is *not* equipped with a microphone but has connected headphones. A malware installed on the computer, can turn the headphones into a microphone for eavesdropping.

Disabled microphones. In this scenario, the computer may be equipped with a microphone, which at some point was disabled, muted (with an 'off' button), or taped (e.g., in laptops [4]). This typically occurs when the user wants to increase security and ensure a conversation's confidentiality. In these cases the malware provides the ability to record using the headphones.

The rest of this paper is structured as follows: Technical background is provided in Section 2. Section 3 discusses design and implementation. Section 4 describes the evaluation and experimental results. Section 5 presents related work. Countermeasures are discussed in Section 6. We conclude in Section 7.

2. Technical Background

In this section we provide the technical background necessary to understand the attack at the hardware, chipset, and OS levels.

The fact that speakers can be used in reverse mode and hence, can function as microphones, has been known for years and is well documented in professional literature [1] [5]. However, this alone doesn't raise a security concern, since it requires that a speaker be intentionally plugged into the microphone jack (an input port) in order to play the role of a microphone.

A glance into the security related issues of such a 'speaker-as-microphone' scenario can be found in a partially declassified document released by the NSA in 2000, in response to an appeal of an earlier Freedom of Information Act (FOIA) request. The document is a guide to the installation of system equipment that takes into account red/black security concerns. It contains the following paragraph:

*"In addition to being a possible fortuitous conductor of TEMPEST emanations**, the speakers in paging, intercom and public address systems can act as microphones and retransmit classified audio discussions out of the controlled area via the signal line distribution.** This microphonic problem could also allow audio from higher classified areas to be heard from speakers in lesser classified areas. Ideally. Such systems should not be used. Where deemed vital, the following precautions should be taken in full or in part to lessen the risk of the system becoming an escape medium for NSI."* (NSTISSAM TEMPEST/2-95, RED/BLACK INSTALLATION [6] [7]).

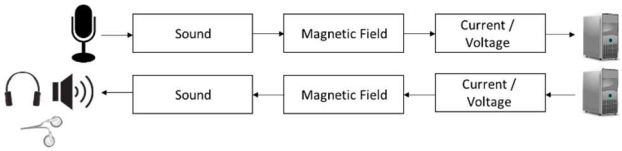


Figure 2. Basic illustration of audio recording and playing, demonstrating that the speaker and microphone are inverses of each other.

2.1 Speakers, headphones, earphones, and earbuds

Dynamic microphones are the inverse of dynamic loudspeakers. In the former, sound pressure variations move a membrane attached to a coil of wire in a magnetic field, generating an electric current/voltage. The inverse happens with loudspeakers: the electric voltage associated with a sound drives an electric current through a coil in the magnetic field, generating a force on the coil and moving the membrane attached to it, producing sound in the air (Figure 2). In fact, in simple intercom systems the same mechanism is used as either a microphone or loudspeaker.

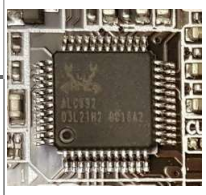
Note, however, that the reversibility principle poses a limitation: the speaker must be passive (unpowered), without amplifier transitions. In the case of an active (self-powered) speaker, there is an amplifier between the jack and the speaker; hence, the signal won't be passed from the output to the input side [8]. Since most modern loudspeakers have an internal amplifier [9], the threat presented in this paper is primarily relevant to headphones and earphones, and not to the loudspeakers typically connected to a PC.

2.2 Jack retasking/remapping

Desktop PCs may have a built-in, onboard audio chip or external sound card. Typical PCs include various analog input and output jacks. Input jacks are used for microphones or other sources of audio stream. The input data is sampled and processed by the audio hardware. Output jacks are used for loudspeakers, headphones, and other analog output playback devices. As noted, the capability of changing the functionality of the audio jacks is referred to as jack retasking or jack remapping.

Intel High Definition (HD) Audio (the successor of AC'97) is the standard audio component on modern PCs. Jack retasking is now part of the Intel High Definition Audio specification [10]. In this standard, the audio chip on the motherboard is referred to as an audio codec. The audio codec communicates with the host via a PCI or other system bus. Realtek Semiconductor Corp. provides the audio codec chip for many motherboard and chipset manufacturers and is thus integrated in a wide range of desktop workstations and notebooks. The most common Realtek codecs in PCs are the multi-channel high definition audio codec series presented in Table 1.

Table 1. Realtek codec chips that support jack retasking

Realtek codec chips (all support jack retasking)	Integrated in	
ALC892, ALC889, ALC885, ALC888, ALC663, ALC662, ALC268, ALC262, ALC267, ALC272, ALC269, ALC3220	Desktop PCs, Notebooks	

As noted in the table, the codec chips listed support jack retasking. In this paper we primarily focus on Realtek codec chips, since they are the most common codecs in PCs. It's important to note that other codec manufacturers support jack retasking as well [11] [12].

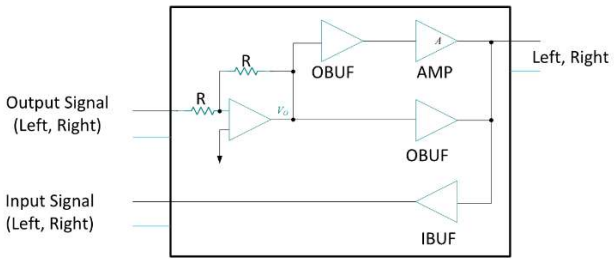


Figure 3. Jack retasking at the hardware level.

2.3 Hardware interface

At the hardware level jack retasking means that the retaskable audio jacks are connected with both ADC (analog-to-digital convertor) and DAC (digital-to-analog convertor) components and hence, can operate as input (microphone) or output (speaker) signal ports.

Figure 3 shows input/output retasking at the hardware level, based on the Realtek design [13]. The schematic diagram of the mic/line physical sockets illustrates two electrical circuits connected to the same physical socket. Choosing the input configuration enables the leg of the IBUF to rise, powering on the buffer and allowing the signal to enter the computer. In contrast, choosing the output configuration enables the legs of the OBUF and the AMP to rise and enables the output buffer and amplifier. This action allows the signal to flow out from the computer to the socket. When buffers are not enabled, signals cannot pass through.

3. Design & Implementation

The HD audio component consists of the controller and codec chips on the HD audio bus. Each codec contains various types of widgets, which are logical components that operate within the codec. Software can send messages (or verbs) to widgets in order to read or modify their settings. Such verbs are sent via the HDA link,

which is a serial interface that connects the audio codec to the host PC. Typical messages to the audio codec are structured as `[NID][Verb][Payload]`, where `NID` is the node identifier (e.g., the widget to operate on), `Verb` is the type of operation (e.g., set configuration), and Payload contains the parameters for the operation (e.g., the configuration parameters).

The HD audio codec defines a number of pin widgets. Each pin, including the audio jack ports, has its own configuration. The configuration includes the jack color, location (rear, front, top, etc.), connection type (in or out), and other properties. For example, in the Realtek ALC892 chip, pins 14-17 (LINE2-L, LINE2-R, MIC2-L, MIC2-R), pins 21-24 (MIC1-L, MIC1-R, LINE1-L, LINE1-R), and pins 35-36 (FRONT-L, FRONT-R) are the analog input and output pins. In the retaskable pins (e.g., 14-17) it is possible to change the default configuration and its functionality from out (e.g., headphone or speaker) to in (microphone) and vice versa. The HDA specification defines the complete codec architecture that allows a software driver to control various types of operations [10].

3.1 Kernel interface

The vendors of audio codec chips, such as Realtek and Conexant, provide kernel drivers which implement the codec's functionality, including retasking, and expose it to the user mode programs. For example, the Realtek driver for Microsoft Windows allows remapping the audio jack via specific values in the Windows Registry (`HKEY_LOCAL_MACHINE\SYSTEM\CurentControlSet\Control\Class\{4D36E96C-E325-11CE-BFC1-08002BE10318}\0000\Settings\`). A guide for how to remap Realtek onboard jacks in Microsoft Windows can be found in [14]. The Linux kernel, a part of the Advanced Linux Sound Architecture (ALSA), exposes an interface that enables the jack configuration; the `hda-jack-retask` tool is a user mode program for Linux that allows the manipulation of the HD audio pins' control via a GUI interface [15].

3.2 Architecture

The architecture of the SPEAKE(a)R malware is shown in Figure 4. The SPEAKE(a)R malware consists of a user level process and a kernel level driver. The user level process manages the set of malware tasks, such as communication, command and control (C&C), persistency, keylogging, and so on. The kernel level driver is in charge of retasking the jacks from output to input and vice versa. The main operational scenario involves a PC that is not equipped with a microphone (or in which the microphone is muted or turned off) but has connected headphones, earphones, or passive speakers.

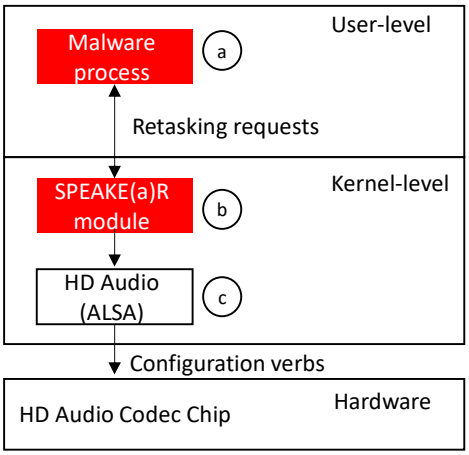


Figure 4. SPEAKE(a)R architecture and components.

In this scenario a malware process runs in the user mode (Figure 4, a), reconfiguring the headphone jack into a microphone jack by sending a retasking request to the kernel module SPEAKE(a)R driver (Figure 4, b). The kernel driver sends the configuration verb to the HD audio codec through the HD audio interface (Figure 4, c), which sends it to the audio codec via the HD audio bus.

Kernel level vs user level. Communicating with the HD audio code chip via the kernel module provides the highest level of stealth, since the malware operations are not exposed to user level monitoring (e.g., anti-virus). Installing a kernel level driver requires root or administrator privileges which can be acquired by stealing credentials or exploiting a privilege escalation vulnerability in the system. However, a kernel level component is not necessary for the implementation of the SPEAKE(a)R malware. A malware can communicate with the HD audio hardware from the user level via command line tools and system APIs in Linux and Windows OSs [16] [17]. These tools and APIs send the configuration verbs via the standard audio driver already installed in the system. The drawback of retasking the audio jacks from the user level is that it is less stealthy. Anti-virus, intrusion detection systems (IDS), and intrusion prevention systems (IPS) can detect the malicious activity, block it, and raise an alert.

Stealth. During normal system behavior the audio output reconfiguration takes place only while the headphones are not in use by the user. To avoid detection, the SPEAKE(a)R kernel module detects when audio output is triggered (e.g., the user is playing music) and instantly reconfigures the microphone jack back into a headphone jack.

Enhancing quality. The infected computer may be equipped with both a microphone and headphones, but the headphones are better positioned for the desired recording, e.g., the headphones are closer to the voice

source, and hence, they can achieve better recording quality. In this case, the headphone jack is configured into a microphone jack, as in the microphone-less case.

4. Evaluation and Experimental Results

Headphones, earphones, and speakers were not designed to perform as microphones in terms of quality and frequency range. Nevertheless, our proposal is rooted in the ability of obtaining a reasonable audio channel when using headphones as a microphone. This section describes a series of experiments performed that are aimed at objectively assessing both the drop in audio quality associated with this arrangement. Specifically, we investigate the relative degradation in audio quality in a variety of controlled experimental setups using either a microphone or a headphone to record human speech. We also investigate the headphones' effectiveness as a receiver in digital communication.

4.1 Experimental setup

We obtained a series of audio quality measurements in order to evaluate audio degradation when the audio is recorded via headphones instead of a standard microphone. To measure speech intelligibility in different experimental setups, we examined a set of pre-recorded sentences used in speech research. More specifically, we used a list of phonetically balanced sentences in American English ('Harvard sentences') as our clean audio reference [18]. The list is comprised of simple phrases containing phonemes (in the same proportion as spoken English), which are often used for standardized development and testing of telecommunication systems, from cellphones to voice over IP. This methodology enables quick and automatic evaluations of speech coding protocols. The list of the Harvard sentences used in our experiments appears in Appendix A.

The actual reference audio used during the experiments was taken from the open speech repository for research [19]. We used an 8 kHz recording of one of the lists by a male and female speaker. The audio was played through commercial multimedia computer speakers (Genius SP-S110) and recorded using an off-the-shelf microphone device (Silverline MM202) and headphones (Sennheiser HD 25-1 II). Several objective speech quality measures were evaluated to estimate the degradation associated with the use of the headphones as described below. The experimental setups varied based on the distance between the computer playing the sound and the recording computer.

4.2 Speech quality

The speech quality measures used in this research were evaluated using the SNReval toolbox [20]. We used the five popular objective speech quality measures described briefly below for each experimental setup. More details and implementation information for these measures can be found in [20].

(1) NIST STNR. Speech to noise ratio, defined as the logarithmic ratio between the speech power and noise power estimated over consecutive 20 msec.

(2) WADA SNR. Waveform Amplitude Distribution Analysis. In this SNR formulation, speech and noise are assumed to follow pre-defined probability distributions. WADA SNR is claimed to be a more stable measurement in terms of bias and variance, compared to NIST SNR.

(3) SNR_VAD. The energy ratio between speech and noise regions designated by some voice activity detection (VAD) procedure.

(4) BSS_EVAL (SAR). Blind Source Separation performance. Evaluated as the source to artifact (SAR) ratio between the clean reference signal and the noise component "separated" from the degraded speech signal

(5) PESQ. Perceptual Evaluation of Speech Quality. Measures speech quality using a psychoacoustic model to compare the reference and degraded speech signals.

The first three measures focus on some version of signal-to-noise ratio (SNR), the ratio between the energy of some speech signal to that of its contaminating noise. In contrast, the last two measures reflect the distortion level of the recorded speech signal with respect to the reference pre-recorded (played aloud) signal and are more directly related to the intelligibility level of the recorded speech.

Table 2. Reference

NIST STNR (dB)	WADA SNR (dB)	SNR VAD (dB)
40.5	49.2	10.1

Table 3. Headphone recordings

Distance (m)	NIST STNR (dB)	WADA SNR (dB)	SNR VAD (dB)	SAR (dB)	PESQ MOS
1	7.5	3.0	2.8	3.7	2.6
3	7.0	-20.0	-7.2	-2.8	2.0
5	6.5	-20.0	-6.1	-5.9	2.0
7	7.8	-2.4	-11.0	-5.5	2.2
9	8.0	-10.4	-20.6	-4.3	2.0

Table 4. Microphone recordings

Distance (m)	NIST STNR (dB)	WADA SNR (dB)	SNR VAD (dB)	SAR (dB)	PESQ MOS
1	29.0	22.6	6.7	8.7	2.5
9	13.8	8.0	4.8	-3.8	2.0

Table 5. ACC coding/decoding

Dis-tance (m)	NIST STNR (dB)	WADA SNR (dB)	SNR VAD (dB)	SAR (dB)	PESQ MOS
0 (Ref.)	39.5	54.4	2.1	12.0	3.5
1	7.5	4.9	-2.0	2.7	2.5
5	6.5	-1.2	-9.4	-5.6	1.9
9	7.8	-1.2	-19.2	-4.5	1.9

Table 2 contains the SNR measurements in decibels (dB) of the reference signal used in the experiments, namely, a sequential recitation of sentences in the Harvard sentences list. Note that SAR and PESQ measures are not included, since this table addresses the pre-recorded reference signal alone.

Tables 3 and 4 show the results for the speech quality measures for headphone and microphone recordings for different recording distances. Table 5 shows the quality degradation of the reference signal and headphone recorded speech after encoding and decoding, reproducing a subsequent transmission of the acquired speech via the Internet. The codec used was the Advanced Audio Coding (AAC) [21], the potential MP3 successor and default codec for YouTube, the iPhone, iPod, and other media devices. This codec has a compression ratio of approximately ten to one. We note that in general, SNR measurements are highly dependent on an accurate segmentation of speech versus noise excerpts. Therefore, in order to optimize segmentation accuracy and consistency across the different setups investigated, voice activity detection was estimated for the reference and not recorded signals. This segmentation was then applied to pairs of reference and recorded signals after they were time-aligned through cross-correlation.

Summary: The results above show a decrease in the SNR when replacing the microphone with headphones for the recordings. Nevertheless, note that the SAR and PESQ indices are much less affected by this change. These indicators, being directly correlated with intelligibility, support our subjective evaluation in which the headphone recordings in our experiments were judged to be intelligible.

4.3 Spectral analysis

In addition to the objective SNR measures presented, we also provide frequency domain graphs corresponding to features used in the above calculations, comparing four transmission setups: (1) microphone, one meter away (from the computer), (2) headphones, one meter away, (3) headphones, five meters away, and (4) headphones, nine meters away.

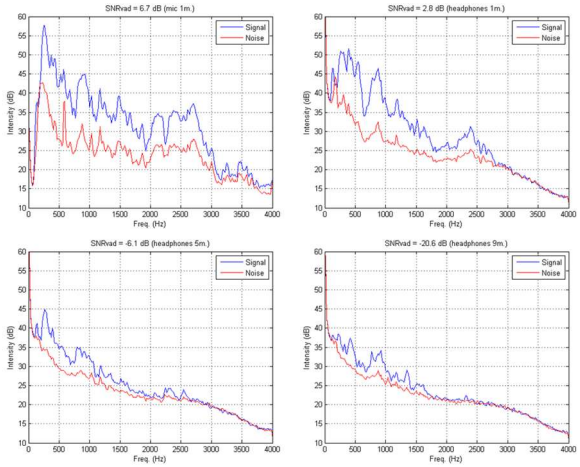


Figure 5. Average signal and noise energy bands for microphone, one meter apart (top left); headphones, one meter apart (top right); headphones, five meters apart (bottom left); and headphones, nine meters apart (bottom right).

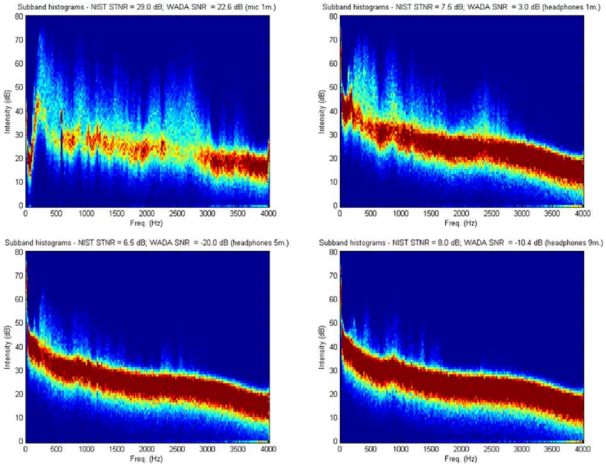


Figure 6. Energy histograms for microphone, one meter apart (top left); headphones, one meter apart (top right); headphones, five meters apart (bottom left); and headphones, nine meters apart (bottom right).

Figure 5 displays average energy in voice-active regions (signal) compared to the average energy in voice-inactive (noise) regions. Note the sharp drop in the SNR in the headphone channel setup for frequencies above around 1500 Hz in comparison with the microphone setup. High frequencies are further compromised as recording distance increases.

Figure 6 contains histograms for the energy levels in decibels for each frequency band. The histograms illustrate the lack of spectrum variability for the headphone spectra in comparison with that of the microphone. Note, as

well, the relatively flat energy distribution for the headphones, especially at higher frequencies.

The results portrayed in the figures and tables indicate that of the speech quality measures utilized the BSS_EVAL (SAR) and SNR VAD are those most correlated with intelligibility. These measures consistently decrease as the distance increases, are far better for microphone recordings (versus headphone recordings), and decrease after AAC coding, as expected. The SAR index is of particular interest, since it is known to correlate, to a certain extent, with subjective ratings thus assessing speech intelligibility. Note, for instance, that the SAR index for a microphone positioned nine meters away from the speech source is in between the indices obtained for headphones located three and five meters apart from the speech source. In addition, note that the codec's impact on degradation does not contribute to a substantial decrease in the speech quality.

4.4 Channel capacity

Thus far we have assessed the speech recording quality attained using headphones as microphones for speech transmission. In this sub-section, we investigate the potential of using the headphone acquired acoustic waves to convey digital information, in terms of channel capacity. We focus on frequencies beyond the hearing range, which can be seen as secure and covert channels for transferring information between two computers.

Channel capacity (C) is a measure of the theoretical upper bound on the rate at which information can be transmitted (in bits per second) over a communication channel by means of signal S. Under the assumptions of additive interfering Gaussian noise (N) and available bandwidth (B (Hz)), the channel capacity can be calculated using the Shannon-Hartley theorem [21]:

$$C = B \log_2 \left(1 + \frac{S}{N}\right)$$

As the formula indicates, the higher the SNR and channel bandwidth, the higher the amount of information that can be conveyed. Note that for large SNRs (S/N >>1) $C \approx 0.33 * B * SNR\ (dB)$. Using this approximation, Figure 7 shows SNR values and respective channel capacity measured for different frequency ranges in our experimental setup, calculated as follows. Similar to the previous experiments, pure sinusoidal tones were played from a source located at different distances from the receiving computer and recorded via the headphones at a 44.1 kHz sample rate. The SNR was measured for consecutive 100 Hz frequency bands up to 22 kHz as the power ratio between the respective frequency tone and the average background noise over the bandwidth.

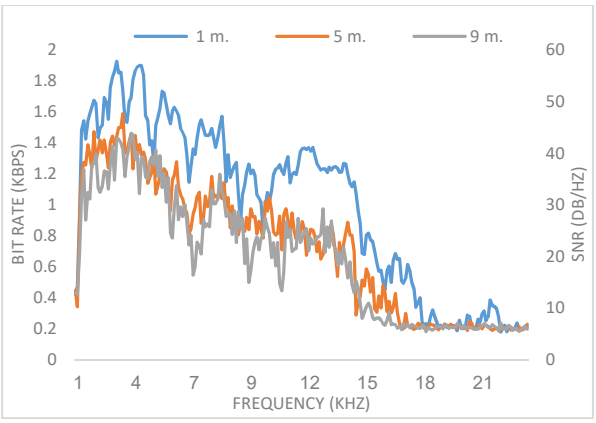


Figure 7. Channel capacity and SNR as a function of frequency.

SNR values were then used to evaluate the channel capacity for each of the 100 Hz bandwidth sub-channels for different transmission distances, using the linear approximation previously described.

Our experiments suggest that headphones turned into microphones have significant potential for covert information transmission, particularly considering that normal human hearing capabilities typically decrease at frequencies over 10 kHz, and large inaudible spectrum regions are available for communication at reasonable rates.

4.5 Improving SNR by combining headphone channels

Headphones and earphones contain two speakers (right and left) which are transformed by SPEAKE(a)R into two microphones (channels). It known that the SNR of sampled audio can be reduced by averaging the outputs of the two channels. Assuming that the noise signals present in different channels are random and thus uncorrelated, combine the headphone/loudspeakers' right and left channels would average out the noise, while enhancing the desired signals which are correlated. Theoretically, the uncorrelated noise sums as a root sum square, resulting in a $\sqrt{2}$ increase, while perfectly correlated signals increase by a factor of 2. This difference yields a 3 dB increase in the SNR. Nevertheless, in practice, we did not succeed in improving the SNR of speech signals acquired through the headphones. We aligned and combined right and left headphone channels, but the overall SNR gain obtained was a marginal 0.1 dB. We believe that due to the headphone channels' proximity, there is a high level of correlation between the channels, and the averaging process is inefficient.

5. Related Work

It is known that PC malware and mobile apps can use a microphone for spying purposes, and many types of spyware with recording capabilities have been found in the wild [22] [23] [24] [25] [26] [27]. In 2015, Google removed its listening software from the Chromium

browser after receiving complaints about potentially exposing private conversations [28]. More recently, Facebook was suspected of (and denied) using a mobile device's microphone to eavesdrop on conversations so it could better target ads [29]. In addition, there are currently many applications sold on the Internet that facilitate the use of microphones and cameras to gather information for surveillance and other purposes [30]. However, such mobile or desktop applications require either built-in or external microphones.

The general principle that an audio speaker is the exact inverse of an active microphone has been well known for years [5], as are the security concerns it raises [7] [31]. Lee et al. suggested turning the computer speaker into a microphone to establish covert communication between two loudspeakers at a limited distance of 10 centimeters [31]. Their work provides comprehensive measurements of different covert acoustic scenarios. However, most of the loudspeakers connected to PCs today have an integral amplifier which prevents passing any signal from output to input, and consequentially, the threat of turning speakers into microphones in modern PCs for eavesdropping hasn't attracted much interest by security researchers, aside from [31].

6. Countermeasures

Countermeasures can be categorized into hardware and software countermeasures.

6.1 Hardware countermeasures

In highly secure facilities it is common practice to forbid the use of any speakers, headphones, or earphones in order to create so-called audio gap separation [32]. Less restrictive policies prohibit the use of microphones but allow loudspeakers, however because speakers can be reversed and used as microphones, only active one-way speakers are allowed. Such a policy was suggested by the NSTISSAM TEMPEST/2-95, RED/BLACK installation guide [33]. In this guide the protective measures state that "Amplifiers should be considered for speakers in higher classified areas to provide reverse isolation to prevent audio from being heard in lesser classified areas." Some TEMPEST certified loudspeakers are shipped with amplifiers and one-way fiber input [34]. Such a protective measure is not relevant to most modern headphones, which are primarily built without amplifiers. Another solution is to implement the amplifier on-board within the audio chipset. Other hardware countermeasures include white noise emitters and audio jammers which offer another type of solution aimed at ruining audio recordings by transmitting ambient sounds that interfere with eavesdroppers and don't allow them to accurately capture what is being said [35].

Table 6. Defensive Countermeasures

Countermeasure	Pros	Cons
Prohibit the use of headphones/earphones/speakers	Hermetic protection	Low usability
Use one way speakers/on-board amplifiers	Hermetic protection	Not relevant to headphones and earphones
Disable BIOS/UEFI audio codec	Easy to deploy	Low usability
Enforce kernel driver policy	Easy to deploy	Can be manipulated by rootkits
Detect the jack retasking	Easy to deploy	Can be manipulated by rootkits
Use white noise emitters/audio jammers	Generic solution	Hard to deploy due to the environmental noise generated

6.2 Software countermeasures

Software countermeasures may include disabling the audio hardware in the UEFI/BIOS settings. This can prevent a malware from accessing the audio codec from the operating system. However, such a configuration eliminates the use of the audio hardware (e.g., for playing music, Skype chats, etc.) and hence, may not be feasible in all scenarios. Another option is to use the HD audio kernel driver to prevent the jack retasking or to enforce a strict jack retasking policy. For closed source OSs (e.g., Microsoft Windows) such a driver must be developed and supported by the various audio codec vendors. In an improved approach, the kernel driver would prevent only out-to-in (speaker to mic) jack retasking, while enabling the use of other types of jack retasking. The kernel driver could also trigger an alert message when a microphone is being accessed, requesting explicit approval of such an operation from the user. In the same manner, anti-malware and intrusion detection systems can employ API monitoring to detect such unauthorized speaker-to-mic retasking operations and block them. A list of countermeasures, along with their pros and cons, is provided in Table 6.

7. Conclusion

Audio playing devices such as headphones, earphones, and simple earbuds can be seen as microphones working in reverse mode: speakers convert electric signals into a sound waveform, while microphones transform sounds into electric signals. This physical fact alone may not pose a security threat, however modern PC and laptop motherboards include integrated audio codec hardware which allows for modification of the audio jacks' functionality (from output to input) in software. In this paper

we examine this issue in the context of cyber security. We present SPEAKE(a)R, a malware that can render a PC, even one without microphones, into a eavesdropping device. We examine the technical properties of audio codec chips and explain why modern PCs are vulnerable to this type of attack. We also present attack scenarios and evaluate the signal quality received by simple off-the-shelf headphones (with no microphone) when used as a microphones. Our results show how by using SPEAKE(a)R, attackers can record human speech of intelligible quality and eavesdrop from nine meters away.

8. References

[1] G. Ballou, Handbook for Sound Engineers, 4th Ed, Taylor and Francis, 2013.

[2] D. Henningsson, "Turn your mic jack into a headphone jack!," [Online]. Available: http://voices.canonical.com/david.henningsson/2011/11/29/turn-your-mic-jack-into-a-headphone-jack/.

[3] [Online]. Available: https://www.electronichouse.com/home-audio/active-vs-passive-speakers-use/.

[4] The Telegraph, "Why has Mark Zuckerberg taped over the webcam and microphone on his MacBook?," [Online]. Available: http://www.telegraph.co.uk/technology/2016/06/22/why-has-mark-zuckerberg-taped-over-the-webcam-and-microphone-on/.

[5] "All Speakers are Microphones," [Online]. Available: http://www.zyra.org.uk/sp-mic.htm.

[6] National Security Telecommunications and Information Systems Security Advisory Memorandum (NSTISSAM), "NSTISSAM TEMPEST/2-95, RED/BLACK INSTALLATION," [Online]. Available: http://cryptome.info/0001/tempest-2-95.htm.

[7] National Security Systems Advisory Memorandum (CNSSAM) , "CNSSAM TEMPEST/1-13 (U) RED/BLACK Installation Guidane," 2014. [Online]. Available: https://cryptome.org/2014/10/cnssam-tempest-1-13.pdf. [Accessed 10 2016].

[8] B. Duncan, High Performance Audio Power Amplifiers, Newnes, 1996.

[9] "wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Powered_speakers#Passive_speakers.

[10] Intel, "High Definition Audio Specification," 2010. [Online]. Available: http://www.intel.com/content/www/us/en/standards/high-definition-audio-specification.html. [Accessed 2016].

[11] conexant, "CX20952 Low-Power High Definition Audio CODEC," [Online]. Available: http://www.conexant.com/wp-content/uploads/2014/06/pb_CX20952.pdf.

[12] IDT, "2-CHANNEL HIGH DEFINITION AUDIO CODEC WITH STAC9202," [Online]. Available: http://www.hardwaresecrets.com/datasheets/STAC9202.pdf.

[13] Realtek, "ALC892," [Online]. Available: http://www.realtek.com.tw/products/productsView.aspx?Langid=1&PFid=28&Level=5&Conn=4&ProdID=284.

[14] "How to remap / retasking Realtek onboard jacks / ports," [Online]. Available: https://www.reaper-x.com/2012/02/13/how-to-remap-retasking-realtek-onboard-jacks-ports/.

[15] D. Henningsson, "Turn your mic jack into a headphone jack!," 2011. [Online]. Available: http://voices.canonical.com/david.henningsson/2011/11/29/turn-your-mic-jack-into-a-headphone-jack/. [Accessed 2016].

[16] "MORE NOTES ON HD-AUDIO DRIVER," [Online]. Available: https://www.mjmwired.net/kernel/Documentation/sound/alsa/HD-Audio.txt.

[17] "High Definition Audio (HD Audio) tool," [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/hardware/dn613936(v=vs.85).aspx.

[18] "IEEE Subcommittee on Subjective Measurements IEEE Recommended Practices for Speech Quality Measurements," *IEEE Transactions on Audio and Electroacoustics* , vol. 17, no. 227-46, 1969.

[19] [Online]. Available: http://www.voiptroubleshooter.com/open_speech/american.html.

[20] [Online]. Available: http://labrosa.ee.columbia.edu/projects/snreval/.

[21] C. Shannon, "Communication in the presence of Noise," *Proc. IRE,* vol. 37, pp. 10-2, 1949.

[22] elaman, "FinFisher IT Intrusion Products," [Online]. Available: https://wikileaks.org/spyfiles/files/0/310_ELAMAN-IT_INTRUSION_FINFISHER_INTRODUCTION_V02-08.pdf. [Accessed 06 11 2016].

[23] BGR, "Former NSA hacker demos how Mac malware can spy on your webcam," 06 10 2016. [Online]. Available: http://bgr.com/2016/10/06/mac-malware-nsa-webcam-patrick-wardle/. [Accessed 06 11 2016].

[24] R. Farley and X. Wang, "Roving bugnet: Distributed surveillance threat and mitigation," *Computers & Security,* vol. 29, no. 5, p. 592–602, 2010.

[25] cnet, "Android malware uses your PC's own mic to record you," 02 2013. [Online]. Available: https://www.cnet.com/news/android-malware-uses-your-pcs-own-mic-to-record-you/. [Accessed 09 2016].

[26] "MOBILE PRIVACY BEST PRACTICES," [Online]. Available: http://www.im.gov.ab.ca/documents/training/OIPC_Mobile_Privacy_and_Security_2014-12-11.pdf.

[27] techdirt, "Smartphone Apps Quietly Using Phone Microphones And Cameras To Gather Data," [Online]. Available: https://www.techdirt.com/blog/wireless/articles/20110417/21485513927/smartphone-apps-quietly-using-phone-microphones-cameras-to-gather-data.shtml.

[28] The Guardian, "Google eavesdropping tool installed on computers without permission," The Guardian, 23 06 2015. [Online]. Available: https://www.theguardian.com/technology/2015/jun/23/google-eavesdropping-tool-installed-computers-without-permission. [Accessed 03 11 2016].

[29] A. Griffin, http://www.independent.co.uk/, 05 2016. [Online]. Available: http://www.independent.co.uk/life-style/gadgets-and-tech/news/facebook-using-people-s-phones-to-listen-in-on-what-they-re-saying-claims-professor-a7057526.html. [Accessed 11 2016].

[30] L. Simon and R. Anderson, "PIN skimmer: inferring PINs through the camera and microphone," in *SPSM '13 Proceedings of the Third ACM workshop on Security and privacy in smartphones & mobile devices*, 2013.

[31] E. Lee, H. Kim and J. W. Yoon, "Various Threat Models to Circumvent Air-Gapped Systems for Preventing Network Attack," in *Information Security Applications*, 2015.

[32] a. Blog, "Air Gap Computer Network Security," [Online]. Available: http://abclegaldocs.com/blog-Colorado-Notary/air-gap-computer-network-security/.

[33] R. I. GUIDANCE, "NSTISSAM TEMPEST/2-95," 12 12 1995. [Online]. Available: https://cryptome.org/tempest-2-95.htm. [Accessed 01 07 2016].

[34] [Online]. Available: http://www.cissecure.com/products/tempest-amplified-speaker-fiber.

[35] L. Bellinger, "9 Counter Surveillance Tools You Can Legally Use," independentlivingnews, 11 2013. [Online]. Available: https://independentlivingnews.com/2013/11/12/20397-9-counter-surveillance-tools-you-can-legally-use/. [Accessed 09 2016].

Appendix A

The Harvard sentences used in our experiments

1. Oak is strong and also gives shade.
2. Cats and dogs each hate the other.
3. The pipe began to rust while new.
4. Open the crate but don't break the glass.
5. Add the sum to the product of these three.
6. Thieves who rob friends deserve jail.
7. The ripe taste of cheese improves with age.
8. Act on these orders with great speed.
9. The hog crawled under the high fence.
10. Move the vat over the hot fire.