A Supraliminal Channel in a Wireless Phone Application

Enping Li
Department of Electrical and Computer Engineering
Binghamton University
Binghamton, NY
eli1@binghamton.edu

Scott Craver
Department of Electrical and Computer Engineering
Binghamton University
Binghamton, NY
scraver@binghamton.edu

ABSTRACT

Supraliminal channels are robust public channels based on multimedia content, allowing innocuous key exchange in the presence of an active warden. One approach to supraliminal communication embeds bits in the pseudo-random state used to produce computer-generated special effects in hybrid media such as a videoconferencing session.

In this paper we describe an application for wireless phones whose audio effects betray pseudo-random state and hence potential key information for an innocuous exchange protocol. Implementation of a walkie-talkie application for the Apple iPhone shows that hundreds of bits can be reliably transmitted per effect.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection—*Authentication*

General Terms: Design, Security

1. INTRODUCTION

Emerging forms of multimedia have opened new horizons in covert communication. Video chat software, for example, allows the user to blend natural video with still background images, video effects and computer animations. Computer-generated content and even randomly generated content can be blended with traditional media, straining existing definitions of "natural" cover data.

In [3], the authors noted that any randomly generated artificial component of a multimedia object is a potential opportunity for undetectable steganography. Rather than altering natural data while attempting to preserve its statistics, we can instead control the generation of artificial data to encode a message, essentially implementing a multimedia mimic function [4]. Under certain conditions, we can create artificial content that encodes a message, but which cannot be distinguished from "innocent" artificially generated content that does not encode a message. Emerging forms of artificially structured communication can present opportunities for perfect steganography.

1.1 Supraliminal Channels

A supraliminal channel is a robust public channel that is plausibly deniable as random. Its purpose is to allow key exchange in a hostile environment where exchange of cryptographic material is forbidden. [1] If we have a robust multimedia hash function $m = h(I)$ that maps computer images to bit strings, we can imagine this function as a decoder for a supraliminal channel. To encode one must either be able to compute an inverse image $I \in h^{-1}[m]$, or to choose a natural image and commit to using $h(I)$ as one's message.

One emerging form of supraliminal channel takes pseudo-randomly generated multimedia objects, and communicates data through the pseudo-random state. For example, a channel described in [3] uses computer-generated special effects in videoconferencing software. These special effects are driven by a pseudo-random generator, and given a recording of the special effect one can deduce the pseudo-random driving bits. This is essentially a robust hash function mapping a video stream to a bit stream. The transmitter of the video then tampers with her computer generated special effect, replacing its pseudo-random data with a file of strong ciphertext. The result is supraliminal transmission.

In this paper we consider the same effect for telephone conversations. If a novelty application (which we write) can allow the immersion of computer-generated audio effects, and if those effects can betray the pseudo-random state behind them, then we have a means of innocuous communication that allows key exchange.

2. COMMUNICATING PRNG STATE IN A TWO-WAY RADIO APPLICATION

In [3], a videoconferencing application was fitted with pseudo-random graphic special effects that betrayed their own PRNG state to the recipient of the video. This basic architecture allows robust transmission of a random string, or of any string computationally indistinguishable from a random string. The main advantage of this approach to covert communication is that the "cover" is a stream of i.i.d. uniform bits, which can be completely replaced with strong ciphertext during content generation. Since the cover is statistically very simple, replacing it leaves no statistical artifacts [1]

[1]This assumes the PRNG output was also computationally

MM&Sec'09, September 7–8, 2009, Princeton, New Jersey, USA.
Copyright 2009 ACM 978-1-60558-492-8/09/09 ...$10.00.

We carry this architecture into a two-way radio, or walkie talkie application, that allows two parties to communicate over the Internet in the style of a vintage walkie talkie. This gives us several opportunities to add novelty sound effects for either realism or surrealism; if pseudo-randomly generated, they too can betray their state to a recipient. The embedding block diagram is illustrated in figure 1.

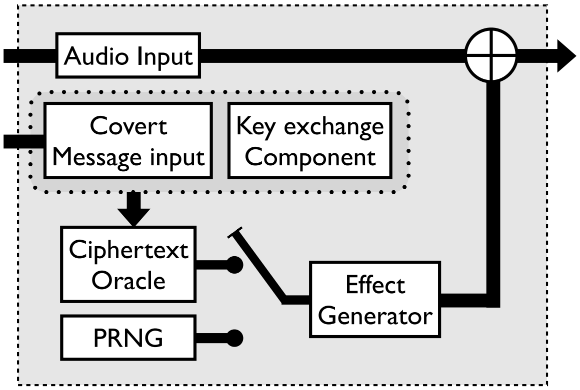

Figure 1: The embedding architecture of our walkie talkie application. An innocent application embeds sound effects generated from a pseudo-random generator; a covert application replaces the PRNG with a ciphertext oracle.

The choice of transmitted ciphertext is governed by a ciphertext oracle and a key management component. Initially, the key management component sends whatever key datagrams are necessary to set up a public key exchange (alternatively, the two devices could share a secret key by some other means.) Upon agreement, each device transmits ciphertext if there is any, else it transmits an encryption of an empty message. A covert text input allows one to feed message data while talking, as shown in figure 2.

Upon receiving data, a similarly enabled application isolates sound effects in the audio stream, estimates its pseudo-random state and attempts to interpret it as key exchange datagrams, and later to decrypt as ciphertext. A curious side effect of supraliminal communication is that the receiver cannot tell if the sender is participating until the moment a meaningful message string is received. If no message is received, it could mean that the communication protocol was derailed by noise, or it could mean that the sender is using an innocent walkie talkie application. There is no way to tell, because any overture to covert communication must be indistinguishable from normal behavior. [1]

2.1 Computer-generated audio effects

For our experiments we have developed a walkie-talkie application for the Apple iPhone, able to record, compress and transmit messages over an Internet connection. We have the capability to add novelty sound effects, which can then be analyzed by the receiving phone.

Plausible sound-effects for our application include:

- *White-noise bursts:* the classic "chhht" sound at the end of each transmission is an obvious choice for transmission with a potentially large data payload. Furthermore, there is a justifiable excuse for a walkie-talkie application to automatically append such a sound after each transmission.

- *Conversational stings.* These can include drum rimshots after one tells a joke, eerie music, laugh tracks, horn sounds, "boo"s and other auditory emoticons.

- *Ambient sound* Continuous ambient background noise has the advantage that does not need to be voluntarily elected, and can transmit a potentially large amount of data.

indistinguishable from random. Replacing the output of a linear-feedback shift register with strong ciphertext is detectable because the former can be identified.

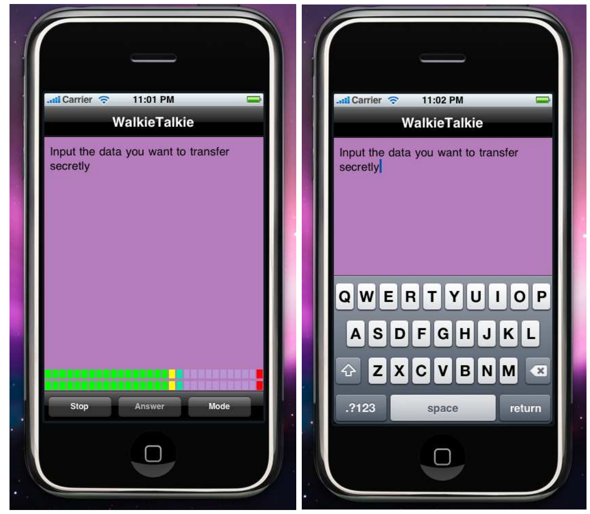

Figure 2: The interface of our walkie-talkie application.

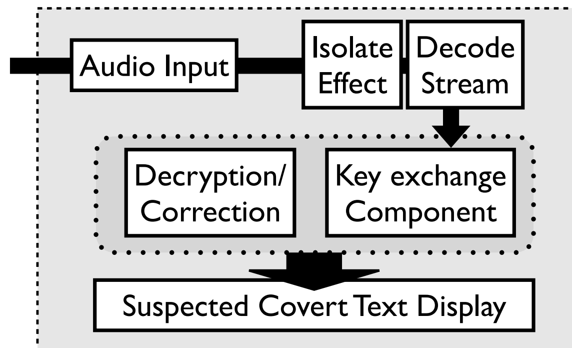

Figure 3: The detection architecture of our walkie talkie application.

For our experiment we have decided to employ walkie-talkie noise bursts, because they are an innocent sound effect, easy to generate and decode, and can be appended to every sound transmission. Ambient sound is a promising means to transmit at large bitrates, but generation is likely to be far more complex. Auditory emoticons seem to hold the least promise because they are short in duration, and would be tiring to users if used with great frequency. It is, after all, insufficient to demonstrate that a supraliminal channel is possible; it must be plausible as a form of communication, even if a novelty.

3. PSEUDO-RANDOM TRANSMISSION

A sound-effect must be generated from a PRNG in a manner that communicates bits. We want to be able to communicate on the order of hundreds of bits per transmission; for public-key cryptography each party needs to transmit a payload of thousands of bits altogether. Our approach is to start with a purported pseudo-random number generator which is instead a file of strong ciphertext, or key exchange datagrams carefully chosen to be indistinguishable from strong PRNG output.

Successful communication requires that we solve two problems of plausible deniability:

1. The random data cannot have any structure that indicates that it is a message; the generation algorithm will be completely known to the adversary, and it must be a plausible algorithm *a priori*. An algorithm that just happens to allow the transmission of an m-bit string is suspicious.

2. The random data must be perfectly recoverable. Recent results show that even under very mild error, key exchange is difficult; furthermore if errors are directed by an active adversary, key exchange can be indefinitely derailed by a very mild noise budget [2]. This is due to the fact that the supposedly pseudo-random data must have no observable structure, and so error correction is usually not possible.

Thus we want a straightforward algorithm to generate a sound effect from a source of random bits, such that the generation is reversible, the algorithm is plausible, and

3.1 Direct sample injection

A simple method to pack random data into a noise sequence is to use the data directly. This renders the data as a playable audio file. This can be sensitive to compression, filtering and distortion of the time domain, depending on the compression codec used.

3.2 Biorthogonal signals

One method is generating Gaussian noise as a sum of biorthogonal signals. Let $\{s_k \in \{\pm 1\}^n\}_{k=1}^m$ be a collection of n-sample biorthogonal signals, and $S = \sum_{i=1}^m (2b_i - 1)s_i$ be their sum. This approximates white Gaussian noise. Information can be encoded as bits b_i, and detected in the decompressed audio by normalized correlation.

In the absence of extra noise, each waveform is impeded by the remaining $m-1$ signals, whose sum is approximately Gaussian with per-sample variance of $m-1$. When normalized by the signal energy, the correlation $(s_k \cdot S/n)$ is approximately Gaussian with unit mean and variance m/n. We want the entire string to be detectable without coding. This gives us a detection probability

$$P_D = \Phi\left(\sqrt{\frac{n}{m-1}}\right)^m$$

Or, solving for the required sample length,

$$n = (m-1)\Phi^{-1}\left(\sqrt[m]{P_D}\right)$$

...so that the message payload determines the number of samples needed to achieve a given failure probability. The samples can then be split among multiple sound effects.

The disadvantage of this approach is that the randomness is relatively contrived, and the structure of biorthogonal signals is evident.

3.3 Gaussian phase encoding

A standard method for generating Gaussian random data is to generate a pair (X, Y) uniformly over the unit disc, and then forming this into a Gaussian pair (Z, W) by stretching its magnitude nonlinearly:

$$(Z, W) = (X, Y)\sqrt{\frac{-2\log(X^2+Y^2)}{X^2+Y^2}}$$

In general, a vector of ni.i.d. Gaussian random variables has a direction that is uniformly distributed over the n-dimensional sphere.

This suggests the following approach to embedding data in a sequence of $2nm$ i.i.d. Gaussian variables:

1. For $j = 0 \cdots n-1$:
   (a) Let $X = \sum_{i=0}^{m-1} X_{2mj+i}$
   (b) Let $Y = \sum_{i=m}^{2m-1} X_{2mj+i}$
   (c) Let $\theta = \pi + \cos^{-1}(X/\sqrt{X^2+Y^2})\mathrm{sgn}(Y)$. θ should be uniformly distributed over $[0, 2\pi)$.
   (d) The decoded b-bit message $P = 2^b\theta/2\pi$.
   (e) To change the message string to a desired value P', let $\phi = (P' - P)2\pi/2^b$, and compute

$$\begin{pmatrix} X^*_{2mj+i} \\ X^*_{2mj+m+i} \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} X^*_{2mj+i} \\ X^*_{2mj+m+i} \end{pmatrix}$$

4. RESULTS

Our walkie-talkie application encodes data using the iPhone ima4 codec at a mono 10000 sample rate.

4.1 Gaussian phase encoding and value packing

Over 200 0.5s bursts, simple Gaussian phase estimation gives us a proper decoding rate of 0.95 if we attempt to extract one bit per pair, or 0.5 bits per sample. For two bits per sample we get detection rate of 0.93, and 0.83 if we attempt to extract three bits per pair.

Directly packing random samples as noise, there is similar error attempting to decode. Simply extracting the sign of each noise sample gives us a detection rate of 0.96. Since this extracts one bit per sample, it appears to be superior to phase detection, and far less complex.

4.2 Biorthogonal signalling

In the experiment, the sampling rate we chose is 10000Hz, and we added 2500 audio samples per transmission as the sound effect. As shown in Fig. 4 the detection rate decreases monotonically as the number of message bits m increases, while the transmission rate achieves the maximum value of 207 bits per transmission, when m falls within the range of $[236, 246]$ bits, which is shown in Fig. 5.

Overall, our embedding rate is maximized in the simple case of generating a white noise sound effect by storing the random bits directly as samples. This allows transmission of kilobits per sample with a small but macroscopic error rate; neither method of robust communication substantially reduces or eliminates this error.

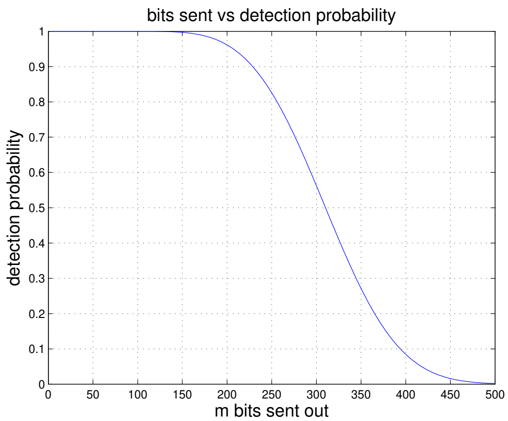

Figure 4: Detection probability per bit, versus number of bits encoded in a 2500-sample effect.

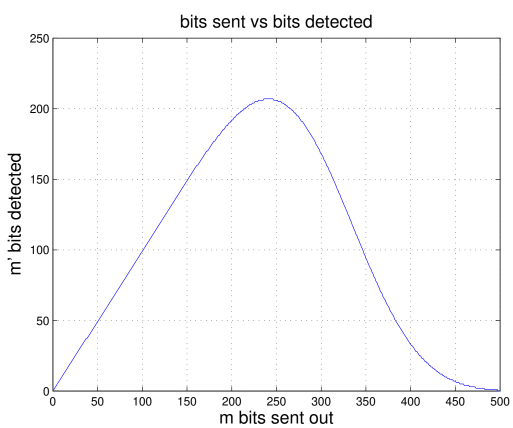

Figure 5: Successful transmission rate in a 2500-sample effect.

5. DISCUSSION AND CONCLUSIONS

We have developed a simple phone application that can smuggle pseudo-random state through innocuous sound effects. Our transmission rate can be thousands of bits per effect if we can accommodate for an error rate on the order of 0.05. However, if we are not allowed to use any coding, a more robust transmission approach gives us transmission on the order of hundreds of bits per effect, requiring many more sound effects to accomplish a key exchange.

As discussed in [3], perfect communication is necessary to survive an active warden who always directs noise to the components of the signal that cause the most harm. A less intelligent active warden, or any mindless noise process, can still inflict considerable harm on a key exchange datagram. This is due to the restriction that any public transmission must look random, and cannot have any overt data correction. On the other hand, once a secret key is agreed upon, error correction can be covertly employed simply by coding prior to encryption in either output-feedback or counter mode.

Our results show that data rates of hundreds to thousands of bits per sound effect are possible, but not yet at a level of reliability needed for key exchange. To achieve this level of reliability, and at the data rates necessary for practical key exchange, we should instead focus on ambient sound effects that can operate continuously during audio transmission.

6. REFERENCES

[1] S. Craver. On public-key steganography in the presence of an active warden. In *Information Hiding, Second International Workshop*, pages 355–368. Springer, 1998.

[2] S. Craver, E. Li, and J. Yu. Protocols for data hiding in pseudo-random state. In *Proceedings of the SPIE, Multimedia Forensics and Security XI*, volume 7254, February 2008.

[3] S. Craver, E. Li, J. Yu, and I. Atakli. A supraliminal channel in a videoconferencing application. In *Information Hiding: 10th International Workshop, IH 2008, Santa Barbara, CA, USA, May 19-21, 2008, Revised Selected Papers*, pages 283–293, Berlin, Heidelberg, 2008. Springer-Verlag.

[4] P. Wayner. *Disappearing cryptography: information hiding : steganography and watermarking(Second Edition)*. Morgan Kaufmann, 2002.