SPEECH CRYPTANALYSIS

S Sridharan*, B Goldburg*, E Dawson**

* Centre for Signal Processing Research
School of Electrical & Electronic
Systems Engineering
Queensland University of Technology

**School of Mathematics
and Information Security Research Centre
Queensland University of Technology

ABSTRACT

The security of frequency domain analog speech scramblers is investigated. It is shown that a vector codebook similar to that deployed in vector quantization of speech can be used to attack the speech scramblers even under the stringent condition that no section of the original speech is available to the attacker and that the encryption key of the system is varied frequently in a random and unknown manner. Subjective as well as objective results demonstrating the success of the attack are given.

1. INTRODUCTION

Cryptanalysis is the process by which one attempts to recover useful information from an encrypted signal. The strategy employed depends entirely on the nature of the signal under attack. For example attacks on written text ciphers can be based on parameters such as the letter frequency Carroll (1988). For speech it is necessary to find other characteristics which are able to be used by the cryptanalyst. The shape of the speech time envelope or the spectral content of a segment of speech, for example, may give clues which will aid the cryptanalytic process. Another factor to consider is the type of speech scrambling employed. A time domain scrambler will obviously require a different attack strategy to a frequency or transform domain scrambler.

The simplest form of attack is one in which the original speech (plaintext) and corresponding scrambled speech (ciphertext) are available. Such an attack is termed a known plaintext attack. In the rest of the text the terms "original speech" and the term "plaintext" will be used interchangeably, as well the terms "scrambled speech" and "ciphertext". Generally it is not practical to obtain ciphertext and the corresponding plaintext. The attack must be undertaken using scrambled speech only. This is referred to as a ciphertext only attack. A cryptanalyst attempts to discover the key based on the ciphertext alone in order to apply the inverse scrambling process to recover intelligible speech.

In Goldburg (1991), previous work by the authors on the cryptanalysis of time domain scramblers can be found. In the current paper we consider the cryptanalysis of analog speech scramblers which are based on the frequency domain. Specifically we consider the discrete Fourier transform (DFT) based speech scrambler Sakurai (1984), Matsunaga (1991), Sridharan (1991). The technique relies on the distortion of the spectral envelope of speech in order to remove intelligibility from the signal. Several speech encryption systems based on this technique are commercially available. The main contribution of this paper is a cryptanalytic attack on these scramblers which is successful even under the stringent condition that no section of the original speech is available to the attacker and the encryption key is varied frequently. Attacks on the DFT based scrambler are given in Section 2. Known plaintext and ciphertext alone attacks are described for both fixed key and varying key systems. In the case of the ciphertext alone attacks, a novel approach is

given using a frequency domain vector codebook to model the speech spectral variation. Subjective as well as objective tests results which demonstrate the success of the proposed attacks are given.

2. CRYPTANALYSIS OF DFT BASED ENCRYPTION SYSTEM

2.1 Description of Scrambler Operation

A DFT based encryption system as described in Sakurai (1984), Matsunaga (1991) and Sridharan (1991) is an extension of the bandsplitter Beker (1985). Typically, the DFT scrambler operates on speech which has been sampled at an 8 kHz sampling rate. The samples are partitioned into frames of 256 samples giving a frame duration of 32 msec. Application of the DFT to a given speech frame will yield a DFT vector containing 256 real and imaginary spectral components. Spectral components within the transmission channel bandwidth are permuted before the inverse transform returns the DFT vector to the time domain. The scrambler simulated for the purpose of evaluating the proposed attacks, permutes 88 spectral components. The scrambling process is illustrated in block diagram fashion in Figure 1 where T is the transform matrix and P is the permutation matrix Sridharan (1991). It is a frame oriented process because the transform operation is based on a fixed number of time samples. The DFT scrambler is more effective in the destruction of the pitch and formant structure of speech than the bandsplitter. This is because the DFT scrambler permutes a large number of individual spectral components rather than a small number of bands which retain spectral envelope information. For the same reason cryptanalysis is made far more difficult. As in the case of the bandsplitter the permutation may be varied from frame to frame to increase the level of security. A scheme which uses 16 permutations on a 16 frame-multiframe basis is described in Matsunaga (1991). Note however in the following attacks we have not assumed any relationship between permutations, ie. the permutations for the varying permutation case is assumed to be selected in a random unknown manner.

2.2 Known Plaintext Attack

The known plaintext attack on the DFT scrambler requires that a single ciphertext spectral magnitude component be matched to a single component in the plaintext. The matching is now made with regard to the relative magnitude of transform components. The largest component in the ciphertext should correspond to the largest component in the plaintext, for example. The pair consisting of the next largest components in the plaintext and ciphertext vectors would also constitute a match. All matches are recorded so that on completion the matched pairs represent an estimate of the permutation used to scramble the frame. Since the real and imaginary part of each DFT component has been moved together by the scrambling process, only its magnitude is required to determine the permutation. Application of this attack to the DFT scrambler described in Section 2.1 resulted in the **complete recovery** of the scrambling permutation using a **single frame** of ciphertext and corresponding plaintext.

2.3 Ciphertext Only Attack on Fixed and Varying Permutation System

The frequency domain vector codebook Gray (1984) can be used in the cryptanalysis of the DFT scrambler using a portion of ciphertext only. The procedure derives an estimate of the permutation using each of the vectors in the codebook. The permutation giving the minimum mismatch between decrypted and codebook spectra is recorded for each frame. Information about the permutation can be accumulated for all available frames in the fixed permutation case. When all ciphertext frames have been used a final decision about the permutation is made, based on the data acquired. In the case of varying permutation case information from previous frame cannot be used and the speech is decrypted on a frame by frame basis. The details of the process are set out below using a codebook of size L, N DFT components and r frames of ciphertext. Note that $r = 1$ gives the varying permutation case.

1. Consider a codebook of spectral magnitude vectors of length L. Let the components of the ℓth vector D_ℓ be denoted by $D_{\ell i}$ representing the ith spectral component where i =1,...N.
2. Input ciphertext vector X and perform the DFT to obtain the spectral magnitude vector F where components denoted by F_j represent the jth spectral component for j =1,...N.
3. Assume that the ℓ^{th} entry of the codebook is the original speech from which spectrum the ciphertext vector F was derived by permutation of the spectral component in vector D_ℓ.
4. Find the largest transform component in the input speech feature vector F. Denote the position on the frequency axis of this component by i. Find the largest transform component of the ℓth codebook vector. Denote the position of this component on the frequency axis by j.
5. Based on the assumption that the largest component $D_{\ell j}$ in the codebook vector (which is assumed to be the original speech spectral vector) will be mapped to the largest spectral component F_i in the scrambled vector F, we record the pair (i,j) as a recovered permutation assignment.
6. Repeat steps 4 and 5 for the next largest component and so on until all the N components of D_ℓ are assigned to one of the N components of F. Use the recovered permutation (i,j) to descramble the scrambled speech spectral vector F to obtain the unscrambled spectral vector U. Let the components of U be denoted by U_i i=1,..N.

(Note: These following steps are used only for fixed permutation case u : r > 1)

7. Determine the squared error $G_{\ell u}$ between the original vector D_ℓ and the unscrambled vector U using the spectral distance :

$$G_{\ell u} = \sum_{i=1}^{N} \left(U_i - D_{\ell i}\right)^2$$

We use $G_{\ell u}$ as a measure of the ability of the codebook vector D_ℓ to model the unknown plaintext vector.

8. Repeat steps 3 to 7 for all values of ℓ=1...L and find the codebook vector ℓ_{min} which gives the minimum error $(G_{\ell u})_{min}$. The corresponding permutation (j,k) determined in step 5 is stored in an assignment table.
9. Repeat Steps 2 to 8 for all r ciphertext frames.
10. Apply the Hungarian assignment algorithm Llewellyn (1964) to the assignment table to output most likely permutation.

A DFT scrambler permuting 88 coefficients, with a frames size N = 256, was cryptanalysed using the procedure described above where a vector codebook of length L = 80 and dimension 88 was used. The attack was performed on a system using a fixed permutation, as well as one using a constantly varying permutation. In each case 300 frames, corresponding to ten seconds of speech, were used for the attack. Table 1 gives the percentage of components which were placed within two positions of their correct location for a typical speech segment. Subjective intelligibility tests as outlined in reference Jayant (1989) were used to indicate the improvement in intelligibility as a result of the application of these attacks. The results are given in Table 2. As expected, the fixed permutation attack gave more intelligibility (% of digits recognized correctly) but significant intelligibility was obtained in the varying permutation case.

3. CONCLUSIONS

This paper has shown that some of the frequency domain encryption schemes which are commonly used are vulnerable to cryptanalytic attack. The scrambled speech produced by each of the scramblers contains information which is accessible and can be recovered using the techniques proposed in this paper. The proposed attack consists of simple spectral matching techniques, but is powerful in that it can recover intelligible speech from the encrypted speech. This is done on a frame by frame basis and is successful even when the permutation (key) is varied from one frame to the next in a random and unknown manner. In each case it is assumed that the frame boundaries can be located. Our experience with commercial scrambling devices leads us to believe that this is fairly easy to achieve. It is also assumed that the number of samples per frame is known to the attacker. Note however that any time jitter in the sampling, introduced at the receiver, does not affect the performance of the cryptanalytic attack. We have also made the justifiable assumption that any non-linear phase characteristics introduced by the channel is able to be equalised by the cryptanalyst. We have experimented with a number of different male and female speakers with different speech segments and found the performance of our attacks to be consistent under the above assumptions. The performance of the attacks under noisy channel conditions was studied and it was found to degrade gracefully as the SNR was reduced from 30dB to 0dB as demonstrated by Table 3.

REFERENCES

Beker, H. & Piper, F. (1985) **Secure Speech Communications**, Academic Press.

Carroll, J.M. & Robbins,L.E. (1988) "Computer Crypanalysis", Technical Report No. 223, Dept. of Computer Science, The university of Western Ontario.

Goldburg, B., Dawson, E. & Sridharan, S. (1991) "The Automated Cryptanalysis of Analog Speech Scramblers", **Abstracts of EUROCRYPT 91**, pp. 203-207.

Gray, R.M. (1984) "Vector Quantization", **IEEE ASSP Mag.**, pp 4-29.

Jayant, N., McDermott, B., Christensen, S. & Quinn, A. (1989) "A comparison of four methods for analog speech privacy", **IEEE Trans. on Communications**, Vol. COM-29, No. 1, pp. 540-547.

Llewellyn, R.W. (1964) **Linear Programming**, Hold, Rinehart and Windston.

Matsunaga, A., Koga, K., & Ohkawa, M. (1989) " An analog speech scrambling system using the FFT technique with high level security", **IEEE Journal in Selected Areas in Communications**, vol. 7, pp. 540-547.

Sakurai, K., Koga, T. & Muratani, T. (1984) "A speech scrambler using Fast Fourier Transform Techniques", **IEEE Selected Areas in Communications**, vol SAC-2, No. 3, pp. 434-442.

Sridharan, S., Dawson, E. & Goldburg, B. (1991) "A Fast Fourier Transform Based Speech Encryption System" to appear in **IEE Proceedings I Communications, Speech and Vision**, Vol. 138, No. 3, pp.215 - 223.

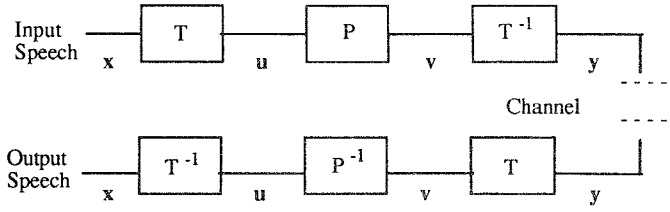

Figure 1
DFT Scrambler

TABLES

Speech Segments Compared	% of Components placed within 2 positions
Original and Scrambled	3.7%
Original and Recovered (Using Varying Permutation)	8.9%
Original and Recovered (Using Fixed Permutation)	23%

Table 1
Results of cryptanalysis DFT Scrambler

	Intelligibility score after attack	
Intelligibility	fixed permutation r = 300 frames	varying permutation r = 1 frame
28.7%	80.5%	49.5%

Table 2
Intelligibility scores for cryptanalysis on DFT based scrambler

SNR (dB)	% of components within 2 of the original post
no noise	13.7
20	13.3
10	12.5
5	7.9
0	6.7
-5	5.3

Table 3
Noise Performance of the Cryptanalysis DFT Based Scrambler-Varying Permutation