

An Eigensystem Approach to Public Key Encryption

Seenu S. Reddi

ReddiSS@AOL.COM

February 25, 2003

Introduction:

Since the introduction of public key encryption by Diffie and Hellman [1], there have been many proposals including RSA [2, 3, 4]. Since the concepts are well discussed in an elementary fashion by young Flannery [5], we refer the interested reader to her book or the resources available on the internet for an introduction and explanation of the basics. The objective of this paper is to introduce a system based on matrix eigensystems for encryption. Most of the existing approaches tend to be number-theoretic and involves computation of long sequences of digits (typically in 100s and in the future may be 1000s) requiring special software or hardware. From an engineering and practical point of view, floating point implementation is preferred since this is readily available and more economical to use. This paper presents a scheme whereby matrices with prescribed eigenvectors and eigenvalues are generated in a systematic manner and public key encryption accomplished using these matrices. Any attack on a system using such encryption involves eigen-analysis and inversion of matrices with a complexity of $O(n^3)$ where n is the order of the matrix. We do not claim that our system is the most advantageous (an advantage of internet publication) but that it is an alternative to existing approaches based on number-theoretic concepts. We conclude the paper with a note on future research indicating consideration of unsymmetric matrices. One may note the unorthodox (or unconventional) presentation of the paper but basically the choice is made in order to present the concepts for the engineer as well as the programmer.

Notation:

Encoding Matrix: \mathbf{M}_E of dimensions $n \times n$ (Published)

Decoding Matrix: \mathbf{M}_D of dimensions $n \times n$ (Private)

Published Matrix: \mathbf{M}_P of dimensions $m \times m$, $m \geq n$ (Published)

Cloaking Matrix: \mathbf{M}_C of dimensions $m \times m$, $m \geq n$ (Guarded)

Inverse Matrix: \mathbf{M}_{inv} of dimensions $n \times n$ (Guarded)

Orthogonal Eigenvectors: ξ_i , Randomly generated $n \times 1$ column vectors (Guarded)

Encoding Eigenvalues: λ_i , $i = 1 \dots n$ - Randomly generated values (Guarded)

Decoding Eigenvalues: μ_i , $i = 1 \dots n$ - Randomly generated values (Guarded)

Introduced Rank Deficiency: R , $R < n$ (Published)

Vector Selection List: $VSL = \{i_1, i_2, \dots, i_S\}$, $S = n - R$ (Published)

Constrained Positions List: $CPL = \{i_1, i_2, \dots, i_R\}$ (Published)

Constrained Values List: $CVL = \{V_1, V_2, \dots, V_R\}$ (Published)

Encoding Multiplicative Parameter: E (Published)
Decoding Threshold Parameter: T (Private)

Generation Mechanism:

Generate n orthogonal vectors consisting of random elements from a distribution such as normal, uniform or other distributions. We choose normal distribution with unit variance and zero mean and Cauchy-Schwarz procedure for orthogonalizing the vectors. Generate eigenvalues $\lambda_i, \mu_i, i = 1, 2, \dots, n$ by selecting from a random distribution with specified variance and a non-zero mean. Compute the encoding matrix \mathbf{M}_E :

$$\mathbf{M}_E = \sum \lambda_i \xi_i \xi_i^T, i = 1, 2, \dots, n$$

and the decoding matrix \mathbf{M}_D :

$$\mathbf{M}_D = \sum \mu_i \xi_i \xi_i^T, i = 1, 2, \dots, n$$

Generate the published matrix \mathbf{M}_P by choosing a cloaking matrix \mathbf{M}_C whose dimensions are larger than that of \mathbf{M}_E and imbed the values of \mathbf{M}_{inv} :

$$\mathbf{M}_{inv} = \sum \lambda_i^{-1} \mu_i^{-1} \xi_i \xi_i^T, i \in VSL$$

Encoding Mechanism:

Convert the ASCII text to be encoded to a sequence of their eight-bit values multiplied by the encoding parameter E. Arrange this sequence into n x 1 vectors by selecting n – R text symbol values and R values from the list CVL. Arrange the R values into column positions indicated by the position list CPL. Multiply the n x 1 vectors with the encoding matrix \mathbf{M}_E and transmit the resulting vectors in double precision (64-bit) or single precision (32-bit) floating point format.

Decoding Mechanism:

From the published matrix \mathbf{M}_P and the private cloaking matrix \mathbf{M}_C , generate the (pseudo) inverse matrix \mathbf{M}_{inv} . Multiply the incoming encoded vectors with the inverse matrix to get intermediate values. If there is introduced rank deficiency, use eigenvectors $\xi_i, i \notin VSL, CPL$ and CVL to account for the deficiency and generate the decoded vectors. Use the threshold parameter to obtain the unencoded transmitted values.

Example:

An example will illustrate the coding and decoding procedures. Assume n = 4 and the following eigenvectors are generated from a Gaussian distribution with zero mean and unit variance:

$$\begin{aligned}\xi_1 &= [-0.82534, -0.34848, 0.16027, -0.41434]^T \\ \xi_2 &= [0.07615, 0.51901, 0.80512, -0.27678]^T \\ \xi_3 &= [0.55923, -0.59962, 0.14309, -0.55429]^T \\ \xi_4 &= [-0.01638, -0.49964, 0.55283, 0.66669]^T\end{aligned}$$

$$\begin{aligned}\lambda_1 &= 4.5, \lambda_2 = 2.1, \lambda_3 = 5.3, \lambda_4 = 3.2 \\ \mu_1 &= 3.9, \mu_2 = 3.1, \mu_3 = 2.4, \mu_4 = 7.7\end{aligned}$$

$$\mathbf{M}_E = \begin{bmatrix} 4.73589 & -0.373773 & -0.0713668 & -0.18320600 \\ -0.373773 & 3.816580 & -0.712441 & 1.04367000 \\ -0.0713668 & -0.712441 & 2.56335 & -0.00774338 \\ -0.183206 & 1.043670 & -0.00774338 & 3.98410000 \end{bmatrix}$$

$$\mathbf{M}_D = \begin{bmatrix} 3.42724 & 0.502453 & -0.203498 & 0.440322 \\ 0.502453 & 4.0938 & -1.25522 & -1.64944 \\ -0.203498 & -1.25522 & 4.51207 & 1.69782 \\ 0.440322 & -1.64944 & 1.69782 & 5.06686 \end{bmatrix}$$

$$\mathbf{M}_{inv} = \begin{bmatrix} 0.0642912 & -0.00390272 & 0.00817153 & -0.00812121 \\ -0.00390272 & 0.0765637 & 0.0542606 & 0.0122902 \\ 0.00817153 & 0.0542606 & 0.102646 & -0.0442497 \\ -0.00812121 & 0.0122902 & -0.0442497 & 0.0457037 \end{bmatrix}$$

Assume introduced rank deficiency $R = 1$ and $VSL = \{1, 2, 3\}$. Let $CPL = \{4\}$ and $CVL = \{0\}$. Also let the phrase to be encoded is "Public" and the encoding parameter is 2. Sequence vectors $S_1 = [2*80 \ 2*117 \ 2*98 \ 0]^T$ and $S_2 = [2*108 \ 2*105 \ 2*99 \ 0]^T$ are computed from the ascii values and the encoding parameter. Multiplying with the encoding matrix \mathbf{M}_E we get:

$$\begin{aligned}T_1 &= \mathbf{M}_E S_1 = [656.291 \ 693.639 \ 324.287 \ 213.389]^T \\ T_2 &= \mathbf{M}_E S_2 = [930.328 \ 579.684 \ 342.516 \ 178.066]^T\end{aligned}$$

T_1 and T_2 are transmitted in their floating point format. For our example we ignore the cloaking matrix and assume that \mathbf{M}_{inv} has already been extracted from it (see comments about the cloaking matrix in the next section on implementation). The recipient multiplies T_1 and T_2 by \mathbf{M}_{inv} to get:

$$\begin{aligned}V_1 &= \mathbf{M}_{inv} T_1 = [40.4036 \ 70.7649 \ 66.8445 \ -1.40184]^T \\ V_2 &= \mathbf{M}_{inv} T_2 = [58.9023 \ 61.5255 \ 66.3347 \ -7.44890]^T\end{aligned}$$

Finally these vectors are decoded by multiplying with \mathbf{M}_D :

$$U_1 = \mathbf{M}_D T_1 = [159.809 \ 228.406 \ 202.180 \ 7.459900]^T$$

$$U_2 = \mathbf{M}_D T_2 = [216.007 \ 210.490 \ 197.446 \ -0.664934]^T$$

Rank deficiency of 1 indicates these vectors contain the null space vector $\xi_4 = [-0.01638, -0.49964, 0.55283, 0.66669]^T$. Using the knowledge that the last position must contain the value of 0, we correct U_1 and U_2 to:

$$W_1 = [159.992 \ 233.997 \ 195.994 \ 0.0]^T$$

$$W_2 = [215.991 \ 209.992 \ 197.997 \ 0.0]^T$$

from which the transmitted phrase “Public” can be decoded.

Implementation Concerns and Complexity Issues:

The purpose of the cloaking matrix is to hide the inverse matrix \mathbf{M}_{inv} from attacks and ideally it should be chosen to be twice as large as the decoding matrix and the inverse matrix should be dispersed within this matrix. Decryption procedure can easily recover the matrix but any attack involving matrix analysis will fail because of the dimensional mismatch. Even with the knowledge of \mathbf{M}_{inv} it is a formidable task to recover the decoding matrix \mathbf{M}_D since this involves matrix inversion which has a complexity of $o(n^3)$. Rank deficiency is introduced in \mathbf{M}_{inv} to defeat the attacks and in principle it is possible to have rank deficient matrices for encoding. Typically the matrix orders will be in the order of 10,000 to defeat any possible attacks. At this time, it is not clear what precision is needed for floating point operations. It is conceivable that the existing 64-bit (or Intel’s 80-bit) precision may be more than adequate with proper choice of parameters.

Future Research:

As mentioned before one avenue for research is the precision needed when the dimensions of the matrix are large. Also one needs to examine whether the encoding procedure reveals information such as the occurrence frequency of the symbols and whether a “noisization” preprocessing can “whiten” such frequencies. Presently unsymmetric matrices are investigated for encoding since they are harder to analyze and devise procedures for attacking. This is an internet publication and any comments may be directed to ReddiSS@AOL.COM.

References:

1. Diffie, W. and Hellman, M. E., “New Directions in Cryptography,” IEEE Trans. Info. Theory, IT-22/6, November 1976.
2. Merkle, R. C. and Hellman, M. E., “Hiding Information and Signatures in Trapdoor Knapsacks.” IEEE Trans. Info. Theory, IT-24/5, 1978.
3. McEliece, R. J., “A Public Key Cryptosystem Based on Algebraic Coding Theory”, Deep Space Network Progress Report, JPL, Pasadena, CA, 1978.

4. Rivest, et al., "A Method for ...", CACM 21, 1978.
5. Flannery, Sarah, In Code, Algonquin Books of Chapel Hill, www.algonquin.com, ISBN 1-56512-377-8, 2002.