[A. Carlson, S. R. Mikkilineni, M. W. Totaro and C. Briscoe. (Jul. 19-22, 2022). A Venona Style Attack to Determine Block Size, Language, and Attacking Ciphers, 2022 International Symposium on Networks, Computers and Communications (ISNCC), Shenzhen, China, 2022, pp. 1-6, doi: https://doi.org/10.1109/ISNCC55209.2022.9851716]

A Venona Style Attack to Determine Block Size, Language, and Attacking Ciphers

Albert Carlson[*], **Sai Ranganath Mikkilineni**[†], **Michael Wayne Totaro**[‡], and **Christopher Briscoe**[§].
[*] Computer Science Department, Austin Community College, Austin, TX, USA.
[†] [‡]Intelligent Systems Modelling and Simulations (ISMS) Research Lab, Center for Advanced Computer Studies (CACS), University of Louisiana at Lafayette, Lafayette, LA, USA.
[§] Physics Department, Iona College, New Rochelle, NY, USA.
Email: [*]*albert.carlson@austincc.edu,* [†]*sai-ranganath.mikkilineni1@louisiana.edu,*
[‡]*michael.totaro@louisiana.edu,* [§]*cbriscoe@iona.edu.*

***Abstract*—The One Time Pad (OTP) is regarded as the only “perfectly secure cipher”. It was extensively used during the World War-II and the Cold War by the USSR. However, the OTP was rendered vulnerable when the USSR started to recycle and reuse the already used keys for encryption; this vulnerability was exploited by the USA during the Cold War with the USSR, which lasted well into the 1980s. The exploit used by the USA was codenamed as the ”Venona-Attack”. Nevertheless, there was little to no declassified information available about the technique used in the Venona-Attack. In this paper, we propose an algorithm that may potentially be the crux of the Venona-Attack, and we establish the conditions that would facilitate a Venona style exploit. The proposed exploit from this paper can be use to attack serial ciphers and any cipher that reuses keys. To demonstrate our algorithm, we present results from our preliminary test corpora to establish the limitations of the attack testing with block sizes of 1 and 3, and also countermeasures that can be employed to counteract a Venona style attack.**

***Keywords*—** Venona Attack, One Time Pad, Stream Ciphers, Block Size, Unicity Distance, Matrix, RNG, Polymorphic Cipher Reduction, Peleg Relaxation, Law of Large Numbers, Vernam Cipher, Block Cipher

I. Background

A. *Technical Background*

1) ***One time pad:*** The One Time Pad (OTP) was first described by Frank Miller in 1882 [1], [2], and was later reinvented, and ultimately patented, by Gilbert Vernam. Vernam used the XOR operation to implement the encryption system in the OTP, for which he was granted a patent in 1919 [3]. While Vernam’s initial design for the system was deemed vulnerable due to its use of a **cyclic** key tape for encryption [4], to surmount this vulnerability, a new cipher, and a key pair are selected for encrypting each character in the message. Even though Vernam’s implementation used the XOR operation as the single cipher for encryption, it was later extended to use other encryptions. The extension of the OTP to use as many ciphers as needed has resulted in improved security through the increase in entropy for an encrypted message.

2) ***Cipher reduction:*** The vulnerability of the Vernam cipher was first recognized in the late 1930’s. The root of the vulnerability was best explained by an IBM researcher Horst Feistel in the early 1970s. In conjunction with the observation, Feistel famously noted that at their core, all ciphers are substitution (S) ciphers [5]. Years later, Carlson confirmed Feistel’s statement through his dissertation research [6] by drawing a comparison between substitution (S) block cipher(s) and other block ciphers and also demonstrated the concept of “isomorphic cipher reduction” [6], [7]. The proposed cipher reduction involved substituting an equivalent S cipher in place of other constituent ciphers (like the permutation (P) cipher) in product ciphers, thus, accommodating consolidation of multi-round product ciphers [8] into a single round S cipher using the property of “idempotence” [9]. Isomorphic cipher reduction is a vital phenomenon due to the property of an S cipher - the representational transmutation of patterns during encryption of a message.

3) ***Drawbacks of a substitution (S) cipher:*** It is well established that ciphertexts generated by an S cipher reveal or “bleed” the base patterns/characteristics of the plaintext, thereby failing to obscure them [6], [10]. Thus, we can extract information about the corresponding plaintext from its S-generated ciphertexts. The significant factors contributing to the relationship of these patterns to obscured information are the syntactical rules of a language and the semiotics [11] induced by the author of the plaintext: Every language has rules that dictate the word use, and sentence structure [12], [13]. Additionally, each user has their individual communication habits that translate into patterns that can identify the user [14], [15]. Together, these induced structural bounds on a language result in a probability density function

(PDF) as a function of the characteristic frequency of the language's alphabet and the combination of symbols from the alphabet itself. Such a PDF taken over a large data corpora curated from a language is representative of the language as a whole, due to the law of large numbers (LLN) [16], [17].

*4) **An adversary and the probability density function (PDF):*** The PDF is one form of representation for a language's symbol frequency statistic, and an adversary can use it to decipher a ciphertext. The probability of each letter appearing in a message is reflected in the Shannon measure of entropy [18], [19]. Entropy is an inverse reflection of the amount of information revealed by encountering a symbol in the encryption/decryption stream. The value of entropy ($H(x)$) is defined as,

$$H(X) = \sum_{i=1}^{n} pr(x_i) log_2 \Big(pr(x_i) \Big) \tag{1}$$

and is related to the redundancy of a language, which is defined as,

$$R_E = 1 - \frac{H(x)}{H_{max}(x)} \tag{2}$$

Redundancy (R_E) is the measure of how often a symbol repeats in a language [18]. Shannon experimentally determined $R_E \approx .75$ for English [20]. Each language has redundancy induced by the syntax of the language and the size of the language's alphabet. R_E is not constant for every message composed using a language, but, depends on the content of the message. Furthermore, Langendoen and Postal assert that each person speaking a language uses a unique version of the language with their individual PDF for that language induced through semiotics [11], [15].

Redundancy plays a vital role in the measure of unicity distance. Unicity distance (n) [18] is the minimum/average number of characters from a ciphertext required by an adversary to accumulate substantial information on the language used for the plaintext and the message to achieve successful decryption of the ciphertext. It is defined as,

$$n = \frac{log|K_c|}{R_E log|A|} \tag{3}$$

$|K_c|$ is the keyspace of the cipher used in encryption, and $|A|$ is the size of the alphabet for the plaintext's language. Although unicity distance is typically measured for an entire message, Carlson et al., through their contemporary work in [21], indicated that it has applications and implications localized within a message.

*B. **History of the attack***

In the late 1930s, with the impending World War-II, an increase in the demand to conceal and secure communications between agencies and their respective agents led to the effusion of electronic cryptographic devices. Even though the Vernam cipher [3] or the one time pad (OTP) [18], [22] used by the USSR was highly secure during its early years of use; the cost of producing and distributing the keys for the OTP was relatively high [23]. As a result, the USSR chose to recycle and reuse the previously used keys instead of generating new ones. This reuse of keys by the USSR lasted well into the early 1980s. During that time, due to their volatile relations with the USSR, the US conceived an attack, taking advantage of the reduced entropy in USSR's secure communications induced by the reuse of keys. This attack was code-named as the Venona attack [24], and it allowed the US to read USSR's encrypted transmissions, and accumulate Intel. This collective intelligence was used to affect the US's war plans during the Cold War, among other things.

While the Venona attack was very successful, there is limited declassified information regarding the attack's technique made available. It is known that the crucial step in breaking the encryption of the messages was to determine which of the previously used keys are used for the current ciphertext; however, the details on how that was accomplished are sparse. Therefore, if we can establish the mechanism of such a consequential step in the attack, we will be able to reduce the unsolvable complex decryption problem of an OTP encryption to a complex yet solvable decryption problem.

In the following section we establish the motivation behind our work in this paper followed by the outline and the description of the Venona Attack followed by the preliminary implementation, testing and results, then close with our conclusions.

II. Motivation

While the theoretical proposal for the OTP used proper random keys and was proven to be unbreakable by Shannon in 1949 [18], generating proper random keys is cumbersome, expensive, and impractical; Therefore, pseudorandom keys were used instead.

A stream cipher is a cipher that the OTP loosely resembles and inspired. Stream ciphers are characterized by synchronizing the message and a pseudorandom key stream. Bits from both the data streams are combined using some function, typically a XOR function. While the design of OTP recommends using a key that is longer than the message, the designers of the stream cipher recommended using a convenient fixed-length key to generate the key stream, thus making it more susceptible to decryption by an adversary than the former.

A popular defense against information accumulation using a ciphertext has been to change the cipher/key pair frequently and irregularly. For implementing this particular defense, we use the total number of continuous characters (known as a ”shard”) with the same cipher/key pair as a measure. This defense aims to change the cipher/key pair more frequently than the local unicity distance to ensure that the attacker does not have enough information to break the encryption for that shard. Choice of the cipher and key for a shard should be as random as possible. If the choice is not random enough, it is possible to attack the randomization. This is especially true if the cycle length of the RNG results in repeating the cipher/key choice during the message(s).

Evaluating results from encryption and probabilistic applications requires being able to accept and apply solutions in a fuzzy manner. Peleg, et al, demonstrated such a method, known as Peleg relaxation [25], which states that the desired solution is probably that which maximizes overall probability of being correct rather than maximizing single probabilities. However, no probability in the collection of possibilities may have a probability of zero.

Peer review of ciphers entails the assumption that the characteristics of a cipher being studied are known. This principal is part of the application of Kerckhoffs' [26] and Shannon's [18] statement of “security by obscurity.” This is meant to ensure that a security study's main goal is to study the strength of the obscuring algorithm and the key set. However, how that information is known is never specified. Our preliminary experimental work in this paper attempts to show how the measure of block size and key reuse can be easily determined as part of a Venona Style Attack at a rudimentary level.

III. The Attack

A. *Characteristics accommodating Venona style attacks*

The focus and limitations of the type of cipher targeted by this attack are as follows:

1) *Keys must be cyclically applied* - Keys must repeat in a finite number of applications (here, ciphertexts). This can be either by design or by intentional reuse. Specifically, this method targets OTPs and serial ciphers.
2) *The base cipher must be a block cipher* - This method assumes that the cipher used for encryption is a block cipher, but does not specify block size. The size of the block ($|B|$) can be any size in the range $1 \leq |B| \leq n$. However, it is assumed for simplicity, that the block size is known *a priori*.
3) *Consistent mapping to a single set of alphabetic symbols* - It is assumed that the target alphabet does not mutate or change during the use of the cipher.
4) *No embedded randomization* - There is no randomization applied during the use of the cipher.

B. *Outline of the attack*

The successful decryption of a ciphertext returns the cipher key and the plaintext message. The goal of any decryption attack is to return the initial message, but, not the key, this is because there may be multiple viable keys that return the same message. Thus, finding the key is just a means to an end. The set of keys that would return the same plaintext message when applied to a ciphertext are known as “isomorphic keys” [6], [27]. However, different messages would yield a different set of isomorphic keys.

As with the decryption efforts by an adversary, there needs to be a sufficient corpus of data to attack. The corpus must consist of enough characters in order to meet the unicity distance for all of the required tests. If the practice of changing keys according to the OTP cipher/key rotation is followed, then sequential messages may be concatenated together to form the test corpus. However, messages must be sequential and linear in order for the corpus to be valid.

The attack presented here is a two-part attack: the first part of the attack seeks to find the frequency of key repetition(s) over a corpus, and the second part of the attack then uses that data to decrypt the contents of a message. Given the knowledge gained in part one of the attack, the second part assumes that all ciphers are S ciphers at their core. Thus, the resultant attack is for an S cipher, and the goal is to find an isomorphic key that works for decryption. Therefore, since this type of attack is well studied and presented [10], the second part of the attack will not be presented here.

C. *The algorithm for the attack*

The iterative algorithm of the attack by an adversary with a large enough corpus of encrypted blocks:

1) *Determine the size of the blocks' matrix* - Starting with the the minimum number of characters in a column to achieve n for the suspected cipher (or for the maximum n for the most secure cipher). In subsequent steps, reduce the row size by 1.
2) *Align characters in rows and columns* - Set up the structure of the matrix using the row size to construct the columns of blocks.
3) *Analyze the information down each column* - Establish the PDF for each column of blocks. The exact composition of the PDF for each column need not be identical. Construct the Universal-PDF so that the blocks are arranged by the probability for each block. This representation allows for easy comparison between PDFs for the various columns.
4) *Compare column PDFs* - If the PDFs are approximately the same, then compare the PDF to the

PDF for the suspected language of the message. The standard being sought is

$$\forall i, j | i \neq j \rightarrow pdf(col_i) \approx pdf(col_j) \tag{4}$$

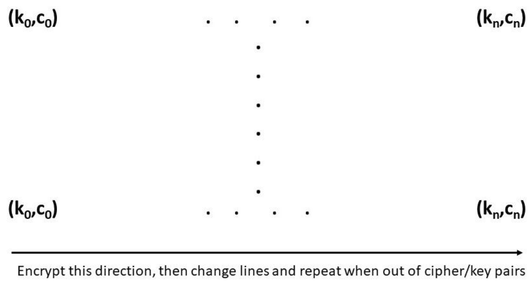
(k$_0$,c$_0$)
. . . .
(k$_n$,c$_n$)
(k$_0$,c$_0$)
. . . .
(k$_n$,c$_n$)
Encrypt this direction, then change lines and repeat when out of cipher/key pairs

Fig. 1. Filling the Data Matrix

If there is no suspected language for the plaintext, then compare the calculated PDFs to known language PDFs to identify possible languages that the plainttext message may have used. This process is similar to using the concept of the "index of coincidence" (I) [28] used to find the key size for XOR, and other ciphers. If the data returned indicates that the λ is wrong, go back the first step. If it does indicate success, try smaller integer multiples for λ. Stop when the length is 1 or the integer multiple of λ is 1.

5) *Solve for each column, simultaneously* - Use the decryption algorithm of choice and use the information found in other columns as *a priori* information in the decryption process of any given column. Relaxation techniques to the variation are applied here.

D. *Definition of the attack*

1) ***Determining the required corpus size:*** Consider a ciphertext of a length sufficient for decryption; this length is calculated based on the unicity distance n.

Now, assume that the plaintext's language and the cipher used for each block of characters ($|A|$) are known. Therefore, it is possible to calculate the value of n for the cipher. Because each λ is considered as a single meta-character in a huge encryption problem, there must be sufficient cycle lengths to meet the n required to accumulate substantial redundancy for plausible decryption by an adversary. Therefore, the minimum size of the corpus required for decryption (s_c) is given by:

$$|s_c| = \lambda n_{c_i} \tag{5}$$

where c_i is the cipher used in a column i, and λ (represents the total number of ciphertexts available in a row) is the total number of columns. If the constituent cipher set is composed of ciphers where the maximum n (denoted by, n_{max}) can use that information to bound the required corpus size. To simplify the calculation the maximum required corpus size ($s_{c_{max}}$) is

$$s_c \leq s_{c_{max}} = \lambda n_{max} \tag{6}$$

2) ***Matricization of corpus:*** Now, consider arranging the corpus data required for decryption in a rectangular matrix as shown in Figure 1. The resulting matrix is of width, λ, and a height of: $\left\lceil \frac{|corpus|}{\lambda} \right\rceil$ (Gives us the total number of Rows).

The main reason for creating such a matrix of blocks (meta-characters), is to enable column-wise working with the data in the matrix. Information in the column(s) has the key to unveil the pattern of reuse of cipher/key pair(s) over the whole cross-section of the corpus; We use this extracted insight to compile the column-wise PDF for the corresponding meta-characters (see Figure 2).

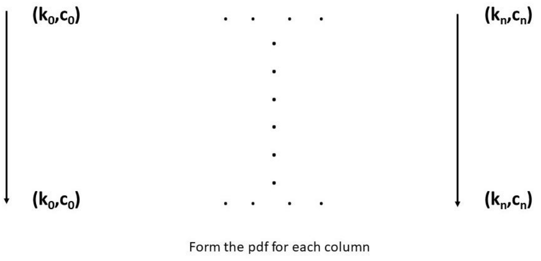

Fig. 2. Evaluation of the Data Matrix

The computed PDF of the information in the column(s) will approximate to the PDF of the plaintext's language, due to the core nature of all ciphers (is S ciphers) [5], and a corpus of appropriate size is representative of the considered language by the LLN [17]. Taken as a group, the right arrangement occurs when all of the columns have a PDF that is approximately to the PDF for a known (or the suspected) value for a particular language. Values for the PDFs in comparison do not have to be precisely identical; that is, some variance is allowed. The variance is preferred to be as minimal as possible, as per the suggestions by Peleg and Rosenfeld [25] and termed as "relaxation." It should be noted that if the size of λ is unknown, then the matrix may have the row size of an integer multiple of λ using the following equation:

$$|row| = i\lambda \tag{7}$$

where $i \in \mathbb{Z}^+$. Testing should continue until the minimal λ where the relaxation occurs.

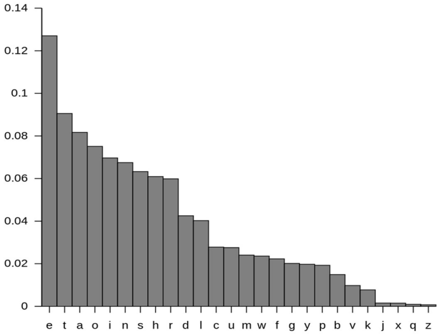

Fig. 3. Probability Density Function for English Language

IV. Implementation, Testing, and Results

A. *Implementation and testing*

Testing the attack consist of two parts: (1) verifying the veracity of language statistics, and (2) applying the attack to encrypted data. The first step is to analyze plaintext in English for both single letters and blocks of letters. Using λ=30 for 47 cases [29] for statistical significance from corpus members of at least 6000 characters. Then, in order to facilitate quick testing S cipher encryption using a rotation cipher with a key of 1 was used. This rudimentary test represents arbitrary ciphers since all ciphers are ultimately S ciphers [5] such a simple encryption can be used without loss of generality. Variation from the true PDF for the language is measures using Euclidean Distance [30], defined as

$$\epsilon = \sum_{i=1}^{|A|} \sqrt{(pr(x_i) - \mu_i)^2} \tag{8}$$

Using a $\lambda = 6$, again without loss of generality and using a rotation of the column number as the key, various matrix widths from 1 - 4 were attempted. The same procedure is also used for blocks of size 3. Choice of the block size is to give a sufficient block complexity and still allow for quick execution time. Test texts are chosen from Project Gutenburg [31] with spaces and punctuation removed, as well as converting capital letters to lower case.

B. *Results*

The results for single byte PDF matching were impressive. As shown in Table I, the variation from the ideal PDF [32], was 0.00164, indicating very small differences. Encryption resulted in the same PDF profiles with a shift in letters corresponding to the shift key. Similar results were found for the 3-gram case. Note that the distance drops for non-trivial factors of $|\lambda|$. This trend needs to be verified with further testing.

TABLE I
Keys for a Given Block Size

No. Bytes	CT/PT	No. Letter Differences	Euclidian Distance	λ	λ_{tried}
1	PT	11	1.09	1	1
1	CT	18	1.7	6	4
1	CT	19	1.68	6	3
1	CT	19	1.68	6	2
1	CT	18	1.7	6	1

In the 3-gram case, the plain text ordered by frequency showed the same characteristics as for the single character case. Relatively, there were more differences in exact 3-gram ordering from text to text, although on average, the highest 15 highly probable 3-grams were identical in order. Some of the differences were the result of "habits" of the author [15]. Some 3-grams did not appear across all cases, resulting in the PDF differences. However, when these non-consistent 3-grams were removed, the PDF remained largely in the same order with respect to the high probable 3-grams. In conclusion, the result is an excellent agreement in 3-gram ordering.

There were some differences in letter ordering, but in general the agreement in PDFs is excellent. Using Peleg relaxation will allow for a directed guess as to the correct PDF mapping.

V. Conclusions and future work

The Venona attack has been written about for many years; however, there is no definitive description in unclassified literature that describes the actual algorithm used by the US-government. Likely the attack consists of many more steps than the basic approach presented here. Certainly, the basic approach is viable to attack any cipher that does not entail a constantly morphing cipher/key pair. The approach presented in this paper makes use of the fact that all ciphers are S ciphers and that the PDF for leaking patterns in the encryption will reach the same PDF as that of the original plaintext language as the corpus for the encryption rises.

A simple matrix was used to align data for easier visualization, manipulation, and computation. Making each row in the matrix equal to the key pad cycle length and then working along the columns ensures that the cipher/key pairs used in the encryption are considered as a set of symbols. When the PDF for each column approximates that of the language, then an integer multiple of λ. Variation between columns is allowable and will probably follow the Peleg relaxation model. Once identified, then decryption follows as a set of simultaneous problems. A method of determining the probability of correct λ lengths was also presented, along with a measure. Determining the required corpus size was also addressed. These measures are based on the

Shannon unicity distance and the anticipated λ of the key pad. Further, it was demonstrated that the corpus does not have to be composed of a single message. Rather, it is possible to use a stream of consecutive messages appended together without any break, so long as the minimum length is met.

Our methodology indicates the need for good RNGs when encrypting using an OTP or polymorphic cipher [6]. Polymorphic RNGs are preferable for this purpose. Obviously it would be best if there were no cycling of cipher/key pairs. Session keys will also defeat this attack if a sufficient corpus of data cannot be assembled. However, if such a corpus can be assembled, then this approach can be used on OTP, polymorphic engines, and serial ciphers. Any cipher with a repeating cipher/key schedule is thus shown to be a serial cipher. These insights verify the given approach and yield insight into the nature of various cipher types and randomization practices. While this might not be the exact mechanism used in the Venona attack, it does suggest that such an approach can be used to implement an effective attack to determine both the cipher/key schedule and the amount of data needed to prosecute such an attack.

Finally, we show that a serial cipher has the same characteristics as an OTP with a repeating key pad. This explains why the approach works for a "perfectly secure" OTP construct. As a result, the importance of having a TRNG or high quality polymorphic RNG is also demonstrated.

Future work requires additional testing of this methodology. Additionally, testing is required on a wide variety of languages to verify the applicability to those languages. More work into polymorphic RNGs is also needed in order to verify that this measure increases security. Testing the same methodology with polymorphic cipher engines is necessary as well.

References

[1] Frank Miller. *Telegraphic code to insure privacy and secrecy in the transmission of telegrams*. C.M. Cornwell, 1882.

[2] Steven M Bellovin. Frank miller: Inventor of the one-time pad. *Cryptologia*, 35(3):203–222, 2011.

[3] G.S. Vernam. Secret signaling system, July 22 1919. US Patent 1,310,719.

[4] NSA. One-time pad https://web.archive.org/web/20140314175211/ http://www.cryptomuseum.com/crypto/otp.html.

[5] Horst Feistel. Cryptography and computer privacy. *Scientific American*, 228(5):15 – 20, 1973.

[6] Albert Carlson. *Set Theoretic Estimation Applied to the Information Content of Ciphers and Decryption*. PhD thesis, University of Idaho, 2012.

[7] Bhaskar Ghosh, Indira Dutta, Shivanjali Khare, Albert Carlson, and Michael Totaro. Isomorphic cipher reduction. 2021.

[8] Uli Maurer and James Massey. Cascade ciphers: The importance of being first. *Journal of Cryptology*, 6(1):55 – 61, 1993.

[9] Richard Wells. *Applied Coding and Information Theory*. Prentice Hall, Upper Saddle River, 1999.

[10] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley and Sons Inc., New York, 2nd edition, 1996.

[11] Daniel Chandler. *Semiotics for beginners*. Routledge, Milton Park, UK., 1994.

[12] Fernando C. N. Pereira and Stuart M. Shieber. *Prolog and Natural-Language Analysis*. Microtome, 1987.

[13] Noam Chomsky. *Syntactic Structures*. Mouton, The Hague, 1957.

[14] Andrew Morton. *Literary Detection*. Scribners, New York, 1978.

[15] D. Terence Langendoen and Paul Postal. *The Vastness of Natural Languages*. The Camelot Press, Ltd., Southampton, 1984.

[16] Frederik Michel Dekking, Cornelis Kraaikamp, Hendrik Paul Lopuhaä, and Ludolf Erwin Meester. *A Modern Introduction to Probability and Statistics: Understanding why and how*. Springer Science & Business Media, 2005.

[17] Sheldon Ross. *A First Course in Probability*. MacMillan Publishing, Inc, New York, 1976.

[18] Claude Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28:656 – 715, 1949.

[19] Paul Garrett. *The Mathematics of Coding Theory*. Pearson/Prentice Hall, Upper Saddle River, 2004.

[20] Claude Shannon. Prediction and entropy of printed english. *Bell System Technical Journal*, 30:50 – 64, 1951.

[21] Sai Ranganath Mikkilineni, Albert H Carlson, Michael W Totaro, Robert E Hiromoto, and Richard B. Wells. An intro to local entropy and local unicity. In *2022 International Symposium on Networks, Computers and Communications (ISNCC): Trust, Security and Privacy (ISNCC-2022 TSP)*, Shenzhen, China, July 2022.

[22] Uli Maurer. A universal test for random bit generators. *Journal of Cryptography*, 5(2):89–105, 1992.

[23] John Earl Haynes and Harvey Klehr. *Venona: Decoding Soviet Espionage in the United States (Yale Nota Bene)*. Yale University Press, 1999.

[24] Venona story - national security agency (nsa)-https://www.nsa.gov/portals/70/documents/about/cryptologic-heritage/historical-figures-publications/publications /coldwar/venona_story.pdf.

[25] Shmuel Peleg and Azriel Rosenfeld. Breaking a substitution cipher using a relaxation algorithm. *Communications of the ACM*, 22:598 – 605, 1979.

[26] Auguste Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX:5 – 83, 161 – 191, 1883.

[27] Albert Carlson, Bhaskar Ghosh, Indira Dutta, Shivanjali Khare, and Michael Totaro. Keyspace reduction using isomorphs. 2021.

[28] William Friedman. Index of coincidence by friedman-https://www.nsa.gov/portals/70/documents/news-features/declassified-documents/friedman-documents/friedman_collection_metadata.xlsx.

[29] R. Lyman Ott and Michael T. Longnecker. *An Introduction to Statistical Methods and Data Analysis 7_{th} edition*. Cengage Learning, 2016.

[30] K.J. Smith. *Precalculus: A Functional Approach to Graphing and Problem Solving*. Jones & Bartlett Learning, 2011.

[31] The Gutenburg Project. Main page-http://www.gutenburg.net. Internet, 2008.

[32] Robert Lewand. *Cryptological Mathematics*. Mathematical Association of America, Washington D.C., 2000.