INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN: 2278 – 0211 (Online)

Strengthening Computer Security By Applying A Graphical Approach –Latin Squares And Informational Entropy

S. Srinivasan
VIT UniversitySCSE,School of computer science and engineering, India
Saniya Chawla
VIT UniversitySCSE,School of computer science and engineering, India
Srishti Chimnani
VIT UniversitySCSE,School of computer science and engineering, India

Abstract:

This paper aims at arriving results which will strengthen the security of computer in simple physical and distributed networks. Latin Squares are used to provide ways of secure access control in a network. A numerical approach is followed for deriving the informational entropy of a network and subjecting it to a condition which is intended to establish maximum security and abate intrusions. A Graphical approach is adapted to represent networks in a form which will ease the analysis about its type, users and links present in the network.

Keywords: Bipartite graph, Latin Square, Security, Informational Entropy

1.Introduction

Security of network systems has been a major concern since ages and holds its importance still. Though the development in encryption technologies have become extensive, it's not credible to say that the networks are completely secure. With the growing number of users joining network, the major being The internet, the need and demand for security will keep on increasing. One of the major issues is secure access of files present in the network of various users. There are many networks which are not centralized and don't have any specific center .Assuming users exist in a group, it's not necessary that users will access files available in respective groups, they would want to access files are outside their group. Also, the groups can be overlapping, in which case, access control for files holds a major priority for any network to protect the privacy for its users.

Graphs have been used to analyze networks and implement a high level of securities to them [ref]. The use of graphs for file securing has also been rigorously studied and analyzed [ref]. In the course of this paper we represent simple and distributed networks as Latin squares via bipartite graphs, where the set of vertices of bipartite graphs are users and files in a group. The Latin square so formed will be used to calculate informational entropy of the considered network, both simple and distributed. We will extend the work presented in[ref] by also considering how accesses can be independent. Through this we will propagate towards maximizing entropy for achieving tightness or high level of security in file accessing.

2.Methodology

2.1.Graphs and their representation: Bipartite Graph to Latin Square

The bipartite graph in mathematical terms can be thought of as a graph where vertices are partitioned into two sets and the edges joining are on vertices of different sets but not within the same set. For our case, we consider the first set as users and second as files in the system. If the user can access a particular file, an edge will exist between the vertex (considered as a user) and other vertex (considered as file). For instance, we take that there are four users in a simple physical network, and can access a set of four files.

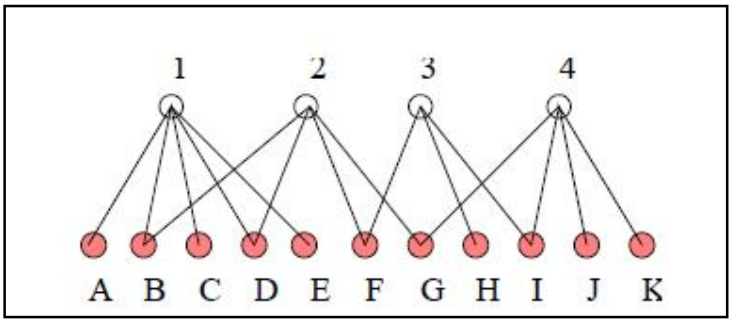


Figure 1

3.Representation- Read Write Accesses

If the every user can access all files, then edges will exist for all vertices in set one to all vertices in set2. This is called as a complete graph and analogically a unrestricted network where there is no restriction on access any file.

Now, this method can be extended to access control for files such as read file and write file. We, represent dotted edge, if a user is allowed only to read a file, and a solid line, if a user is allowed to write a file. If both privileges i.e. read and write are granted to user, multiple edges will exist between the nodes (vertices).

4.What are Latin Squares?

Now we move forward, and convert our network to Latin squares. As seen in the Fig1. , let the x-axis of square represent files and y represent users. A **Latin square** is an $n \times n$ array filled with n different symbols, each occurring exactly once in each row and exactly once in each column.

Now for arriving at results we will consider orthogonal and mutually orthogonal Latin Squares.

A Latin square is called as mutually orthogonal if all elements are orthogonal to one another.

4.1.Number Of Mutually Orthogonal Latin Squares

A Latin square is called as mutually orthogonal if all elements are orthogonal to one another.Orthogonal latin squares are generally hard to come by. There are no orthogonal latin squares of order 2 because there are only two latin squares of order 2 in the same symbols and they are not orthogonal. There are orthogonal latin squares of order 3 as exemplified above. Orthogonal latin squares of order 4 exist but won't be exposed

without a little struggle. Orthogonal latin squares of order 5, or any odd order, on the other hand, are not so hard to find. Let A=(i+j) be the addition table for the integers modulo 2n+1 and let B=(2i+j), entries taken modulo 2n+1. Then A and B are orthogonal latin squares. For, A is a latin square because it is the addition table for the integers modulo 2n+1 and B is a latin square because it is just A with its rows rearranged. To show that A and B are orthogonal, suppose that for some i,j,m,n, the ordered pairs (i+j,2i+j) and (m+n,2m+n) are equal. Then i+j=m+n and 2i+j=2m+n. Subtracting (modulo 2n+1) the first equation from the last, we get i=m, from which it follows that j=n. Hence all ordered pairs from different cells of the two latin squares are distinct and the squares are orthogonal. Here is an example for order 5.

0 1 2 3 4	0 1 2 3 4
1 2 3 4 0	2 3 4 0 1
2 3 4 0 1	4 0 1 2 3
3 4 0 1 2	1 2 3 4 0
4 0 1 2 3	3 4 0 1 2
B	C

There are no orthogonal latin squares of order 6 but it took a long, long time to find this out. Leonhard Euler began to believe that there are no orthogonal latin squares for orders 2, 6, 10, 14, and indeed for all orders of the form 4n+2. In 1900, G. Tarry has proven that no orthogonal squares of order 6 exist thus lending credibility to the Euler's conjecture. 60 years later, in 1960, it was shown by Bose, Shrikhande, and Parker that, except for this one case, the conjecture was false.

The property we will use here is of MOLS (Mutually orthogonal Latin squares). For this we will consider a network, in which the number of nodes are prime or are power of prime, we here onwards refer this condition as Prime Condition. The number of MOLS for such a graph mapped network will be p-1, where p is the number of nodes in a graph/network. The number of MOLS is not yet known is because for our discussion we need number of MOLS, and in general for a graph with n number of nodes, the number of MOLS is not known.

5.Establishing A Relation

As we saw that a network can be represented as bipartite graph, where the users form one group and files another. The graph can be represented as a Latin Square.

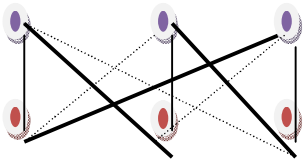
Let us say we have above graph

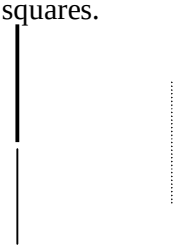
A Square will be : Let the set of edges above be marked by 0.

	A	B	C	D	E	F	G	H	I	J	K
1	0	0	0	0	0	0					
2		0		0		0	0				
3						0		0	0		
4									0	0	0

Table 1

But it violates the property of Latin squares. So, for our problem ,we have considered networks with will satisfy the property of Latin squares.





0,1,2 respectively

The latin square for this graph will be:

	A	B	C
1	0	1	2
2	2	0	1
3	1	2	0

Table 2

But here we will consider that the graph on a network is having prime condition. The edges in the graph, for example say represent read access to files for users. Now we have to get a solution or a optimized way to get a the possible read access to the nodes for minimum conflicts, so that the risk to being attacked or the system going malfunctioned can be minimized. In how many possible ways, can the write access be granted?

Here comes the role of MOLS (Mutually Orthogonal Latin Squares). We first draw the Latin square for the graph, where the edges are denoting the read access . The number of

nodes is p, where p satisfies prime condition. So, number of mutually orthogonal Latin Squares is p-1. Now, here we get can arrive at a clue for , how to grant write accesses. If the number of write access be considered as number of MOLS of that particular graph for the considered network, then we can arrive at a result that the no. of write access can be granted in p-1 ways. This will avoid the multiple edges which existed in the graph with both read and write accesses . Also, In this way the security can be enhanced .

6.Informational Entropy

6.1.What is Informational Entropy

Entropy is a measurement of uncertainty. It is a way to assign a score of uncertainty to a stochastic variable. Entropy is a key component of information theory, a branch of mathematics designed to quantify information. It first came of age in 1948 with the publication of Claude Shannon's paper "A Mathematical Theory of Communication."

6.2.*Shannon's Entropy*

Shannon's classic formula for computing entropy is shown below.

$$H(X) = \sum_{i=1}^{n} p_i \log_2 \left(\frac{1}{p_i}\right)$$

The entropy H of a random variable X with possible outcomes of **{**X1 **…** Xn**}** is the product of the probability of outcome ***i*** times the base 2 logarithm of one over the probability of outcome ***i***. The Greek letter Sigma (∑) indicates summation notation, and as such, the formula is repeated for every possible outcome of i and summed. The overall result is the entropy measured in bits. The base 2 logarithm (also known as the binary logarithm) is most often used to compute entropy because of its close connection with the binary numbering system, and this has become the standard unit of measure when discussing information entropy.

Entropy is an important mark in the information security practitioner's area.

6.3.*Step Towards Maximizing This Entropy*

The first step in our sequence of steps is to consider a number of isolated groups that do not intercommunicate, but which are exposed to an external bath of users (e.g. the Internet), of which certain fraction is hostile. We then ask the question, how does the size of a group affect its vulnerability to attack from this external bath of users? The maximum number of possible groups Gmax, of any size, that one could create on a system of U users is

$$G_{\max} = 2^U - 1. \qquad (1)$$

That is, each group is defined by a binary choice (member or not) on each of U users—but we do not count the group with no members. Groups that do not fall into this basic set are aliases of one another. This number is still huge.There are clearly many ways of counting groups; we begin by considering the problem of non-overlapping groups, i.e. only non-overlapping subgraphs (partitions). Groups are thus (in this section) assumed to be isolated from one another; but all are exposed to the (potentially hostile) environment. Given these conditions, we then ask: Is there a way to decide how many users should be in each group, given that we would like to minimize the exposure to the system?

Let i = 1 . . .G, where G is the number of groups defined on a system. In the following, we will treat G as fixed. Varying G, while maximizing security, generally gives the uninteresting result that the number of groups should be maximized, i.e., equal to the number of users—giving maximal security, but no communication. This is analogous to the similar result in ref. [12]. Let ui be the number of users encapsulated in group i, and fi be the number of files in (accessible to) group i. We say that a link belongs to group i if both its end nodes do. The total number of links in a group is then

$$L_i = u_i f_i, \qquad (2)$$

A graph theoretical model of computer security 9 (see fig. 1) and the probability that a given link is in group i is thus

$$p_i = \frac{L_i}{\sum_{i=1}^{G} L_i}. \qquad (3)$$

Now we can think of putting together a user/file system 'from scratch', by laying down links in some pattern. In the absence of any information about the different groups (other than their total number G) we want to treat all groups equally. For a large system, we can then lay down links at random, knowing that the resulting distribution of links will be roughly uniform. Hence we can (statistically) avoid large groups (and hence damage spreading to a large number of users and files), simply by maximizing the entropy (randomness) of the links. The entropy of the links is

$$S = -\sum_{i}^{G} p_i \ln p_i. \qquad (4)$$

With no constraints, maximizing the entropy leads to a flat distribution,
i.e. that all the pi should be equal in size, or

$$u_i \propto \frac{1}{f_i}, \qquad (5)$$

i.e. the total number of links per group should be approximately constant. This clearly makes sense: if many users are involved, the way to minimizespread of errors is to minimize their access to files. However, this simplemodel does not distinguish between groups other than by size. Since there isno intergroup communication, this is not an interesting result. To introduce a measure of the security level of groups to outside penetration, we can introduce a number ϙi which represents the exposure of the files and users in group i to outside influences. ϙi is a property of group i that is influenced by security policy, and other vulnerability factors. That is, in our model,the ϙi cannot be varied: they are fixed parameters determined by 'outside'factors.

We are thus now looking at a closed model for an open system. We labeleach group i with a positive value ϙi that is zero for no exposure to outsidecontact, and that has some non-zero, positive value for increasing exposure.The value can be measured arbitrarily and calibrated against a scale β, sothat βϙi has a convenient value, for each i. The

average exposure of thesystem is fixed by hϱi, but its overall importance is gauged by the scale β.β may thus be thought of as the 'degree of malice' of the bath of outsiders.We will again maximize entropy to minimize risk. Typically, the proper-ties of the outside bath are not open to control by the system administrator, who first makes the ϱi as low as possible, and then varies the group size, asgiven by the pi, at fixed β. This gives the average exposure:

$$\sum_{i=1}^{G} p_i \epsilon_i = \langle \epsilon \rangle = \text{const.} \qquad (6)$$

However, using a continuum approximation that is accurate for large G, both β and hϱi can be taken to be smoothly variable, and an equivalent procedure is to fix the average exposure hϱi, then vary β. The result is the invertible function β(hϱi), determined by the group sizes pi. We therefore look for the distribution of pi with maximum entropy, i.e. the configuration of minimal risk, that satisfies this exposure constraint, and the normalization constraint Pi pi = 1. We use the well-known Lagrange method on the link-entropy Lagrangian:

$$\mathcal{L} = -\sum_{i}^{G} p_i \ln p_i - A\Big(\sum_{i}^{G} p_i - 1\Big) - \beta\Big(\sum_{i}^{G} p_i \epsilon_i - \langle \epsilon \rangle\Big). \qquad (7)$$

Maximizing $\mathcal{L}$ with respect to p_i, A, β, gives

$$p_i = e^{-A-1} e^{-\beta \epsilon_i} = \frac{e^{-\beta \epsilon_i}}{\sum_{i=0}^{G} e^{-\beta \epsilon_i}}. \qquad (8)$$

Following this simple procedure gives us the result [Eq. 8] that groups with high exposure to attack should have exponentially fewer links than those with less exposure.

But this result has already been derived, so we will extend it further by considering the renyi entropy, which generalizes the shannon's entopy, that has been considered above.

6.4.The Renyi's Entropy

The Rényi entropy of order α, where $\alpha \geq 0$and $\alpha \neq 1$, is defined as

$$H_\alpha(X) = \frac{1}{1-\alpha} \log \left(\sum_{i=1}^{n} p_i^\alpha \right)$$ [1]

Here, $X$is a discrete random variable with possible outcomes $1, 2, ..., n$and corresponding probabilities $p_i \doteq \Pr(X = i)$for $i = 1, \ldots, n$, and the logarithm is base 2. If the probabilities are $p_i = 1/n$for all $i = 1, \ldots, n$, then all the Rényi entropies of the distribution are equal: $H_\alpha(X) = \log n$. In general, for all discrete random variables X, $H_\alpha(X)$is a non-increasing function in α. Now putting this in equation 8, 9, we get

p_i α as the result given above.

As a result the maximized p_i will be the te expression derived, root of α.

Though the maximum will be achieved when alpha tends to 1. But in some networks its not possible to achieve this condition due to high external exposure.

7.Result

A method is proposed to achieve a minimum conflict network by using bipartite and Latin squares by taking the read write access conflicts.

- RESULT 1: The number of ways in which, one can assign write/read access to nodes in a network, given their read/ write accesses is p-1, where p is the number of nodes in a network, that satisfies the prime condition. The risk of conflicts can be minimized and a network can be organized in way, more tightly than the existing one.
- RESULT 2:A result of information entropy is extended to a general scenario of networks, where a already derived result is extended by Renyi's entropy.

8.Conclusion

In this paper we discussed that how a network can be generalized in form of bipartite graph and how their read write access to the files can be represented as form of edges. A possible number of granting access has also been arrived onto. Further informational entropy is discussed and how it can be used to maximize the security. To, be general Renyi's entropy has also been brought into play ,where problem is generalized as whole with a aim of maximizing security , b maximizing the entropy.

9.Acknowledgement

We are very grateful to Professor S. Srinivasan , for guiding us during the course of paper and helping us achieve at useful and valuable results.

9.Reference

1. The following sources proved useful during the course of research:-
2. https://docs.google.com/viewer?a=v&q=cache:Uz9xA3wd1RsJ:journal.univagora.ro/download/pdf/604.pdf+informational+entropy+in+network+security&hl=en&gl=in&pid=bl&srcid=ADGEEShRw3WTKYrfNs0TOfriKWnrdjdDbynAwkPcqcjAQz2xOV3yURxweaReFBXki7CMUJQE6m5hwHUqKjXPxxuuut0lbpY94ECP6-EFhuHZYXDWZOQ9E1GeaRvwaIIvzj6vZd8GqyXS&sig=AHIEtbTinlJQvGxTd5zJw62X-sHIbMDspA
3. http://chemed.chem.wisc.edu/chempaths/GenChem-Textbook/Measuring-the-Entropy-989.html
4. http://en.wikipedia.org/wiki/Principle_of_maximum_entropy
5. https://en.wikipedia.org/wiki/Latin_square
6. http://www.cut-the-knot.org/arithmetic/latin3.shtml
7. http://mathworld.wolfram.com/LatinSquare.html