Identifying the most critical trajectory of the spread of a social engineering attack between two users

A O Khlobystova[1,2], M V Abramov[1,2] and A L Tulupyev[1,2]

[1] St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Theoretical and Interdisciplinary Problems of Informatics,14-th Linia, VI, No. 39, St. Petersburg, 199178, Russia
[2] St. Petersburg State University, Mathematics and Mechanics Faculty, Universitetsky pr., 28, Stary Peterhof, 198504, Russia

Abstract. In this paper, estimates of the probability of success of a multi-pass social engineering attack are considered. The purpose of the article is to identify the most critical path for the spread of a multi-way socio-engineering attack between two users. An approach is proposed for finding the most critical trajectories, an estimate of the probability of success of the passage of an attack on which will be the highest. Formally, the problem reduces to finding a path in the graph, in which the product of the weights of all edges entering the given path is maximal. The use of known algorithms in the solution of the problem was complicated by the fact that the weights of edges of the social graph are estimates of probability. This problem was solved by means of identical mathematical transformations.

1. Introduction

According to [5, 6], the cybercrime rate is increasing from year to year which causes significant losses to companies. The cybercrime nature is also becoming more diversified. Currently, to violate information security of a company malicious people resort not only to search for program and technical system vulnerabilities but to social engineering methods as well, i.e. search for users' vulnerabilities. Thus, the issue of security increase for information system users in face of social engineering attacks appears to be relevant. An important component of this task includes security analysis and monitoring for information system users in face of social engineering attacks.

Based on the number of users involved, social engineering attacks can be one-phase (direct) and multiple-phase implemented through the user chain where the first and final user are not the same people. Some approaches assessing user security under direct social engineering attacks are in sufficient detail stated in [1]. In this article, the issue of user security analysis under multiple-phase social engineering attacks is investigated. The approach assessing user security under multiple-phase attacks and calculation of estimated probability of the attack progression from user to user was presented in [7]. Keep in mind that in [7] when assessing success probability of a multiple-phase social engineering attack, non-directional social graphs are considered, however assessment of probability of the attack progression from user to user in direct and reverse direction can be different. In addition, [7] doesn't consider the issues of search for the most critical attack progression paths. As a rule, there are several possible paths for attack progression from one user to another and progression success probability for each of them will have different values. In this regard, the issue of search for the most critical paths which point higher success probability in attack progression seems to be

relevant. In the context of this issue, one of the tasks concerns determination of the most critical paths for attack progression from one user to another, to solution of which this material is devoted.

2. Formalization of the task to determine the most critical path for social engineering attack progression

The analysis of possible paths for social engineering attack progression is offered to be made based on the direct social graph of the company staff. We will understand the social graph of the company staff as the graph which nodes correspond to the company staff while ribs — to contacting between employees. Formalizing the specified information, suppose there is a graph

$$G = (U, E) \tag{1}$$

where $U = \{User_i\}_{i=1}^{n}$ — a set of nods (users), $E = \{u_i, u_j, p_{i,j}\}_{1 \le i,j \le n, i \ne j}$ — a set of ordered triads with the estimated probability for attack progression from user to user — $p_{i,j}$. Keep in mind that equality of $p_{i,j}$ and $p_{j,i}$ isn't supposed. That means that the probability for attack progression from the first user to the second one can totally differ from attack progression probability that proceeds from the second user to the first one.

In the considered model, estimated probability for successful attack progression from user to user depends on interaction intensity between users. According to [7], it can be calculated as follows

$$p_{i,j} = 1 - \prod_t \left(1 - p_t^{i,j}\right)^{n_t} \tag{2}$$

where $p_t^{i,j}$ is assessment of probability for successful malefactor's social engineering attack to the user using the t route, n_t is the number of interaction episodes., "Like" marks, reposts, etc. are considered as the episodes. In the considered particular case of the model considering the data taken from social networks, $p_{i,j} > 0$, ribs where probability assessment $p_{i,j} = 0$ is excluded from the total social graph.

Thus, the task of search for the most critical paths of multiple-phase social engineering attack from $User_i$ to $User_j$ comes down to the issue of sticking to the elementary path graph (simple, without cycles) between these nods. And the path must be the one that assessing probabilities for transitions from user to user involved there is as extensive as possible. Let us treat assessment of probability for successful multiple-phase social engineering attack which is the multiplication of estimated probabilities for attack progression from user to user and a direct first user attack, as a path length.

3. An approach to the identification of the most critical trajectory of dissemination of the social engineering attack

To facilitate let us consider the graph $G = (U, E')$, where $U = \{User_i\}_{i=1}^{n}$ — a set of nods (users), $E' = \left\{u_i, u_j, \frac{1}{p_{i,j}}\right\}_{1 \le i,j \le n, i \ne j}$ — a set of ordered triads, where number u_i, u_j is associated against each couple of users $\frac{1}{p_{i,j}}$. Let us note that in this case if $p_{i,j} \ge p_{l,k}$ then $\frac{1}{p_{i,j}} \le \frac{1}{p_{l,k}}$. The path length will be evaluated the following way:

$$\frac{1}{p_{ml}} = \frac{1}{p_m} \prod_{i=m}^{l-1} \frac{1}{p_{i,i+1}} \tag{3}$$

where p_{ml} — the assessment of potential of successful passage of the attack from user m to user l, p_m — the assessment of the potential of successful passage of intruders social engineering attack to the user, $p_{i,i+1}$ — an appropriate assessment of potential of attacks spreading to the user through

another one. Thus lets us go to the task of finding a path with the minimum length from the task of finding a path with maximum length.

There is a need of number of transforming to use algorithms of finding minimum length path. According to the main logarithm identity

$$\frac{1}{p_{ij}} = e^{ln^{\frac{1}{p_{ij}}}} \tag{4}$$

Than the path length will be evaluated the following way

$$\frac{1}{p_{ml}} = \frac{1}{p_m}\prod_{i=m}^{l-1}\frac{1}{p_{i,i+1}} = e^{ln\frac{1}{p_m}}\prod_{i=m}^{l-1}e^{ln^{\frac{1}{p_{i,i+1}}}} = \exp\left\{ln\frac{1}{p_m} + \sum_{i=m}^{l-1}\frac{1}{p_{i,i+1}}\right\} \tag{5}$$

The task a heads to searching the path where

$$\sum_{i=m}^{l-1} ln\frac{1}{p_{i,i+1}} \tag{6}$$

is the least among all possible trajectories that start from user m and finish with user l because evaluation of the success of straight social engineering attack on user m — p_m will be equal for all the trajectories.

Thus this task is a standard search of the shortest path in the directed graph without negative weighted edges. Let n is a number of vertices in social graph (a number of staff) and m is a number of directed edges in it. For solving the problem of searching the most critical trajectory on social graph the Bellman-Ford's, Levit's, Floyd-Warshall's and Dijkstra's algorithms, topological sorting and A* were applied [3,4,8]. Bellman-Ford's, Levit's and Floyd-Warshall's algorithms have high computational complexity in context of our task. Solving the problem with the help of topological sorting's algorithm is impossible regarding the conditions of graph's acycling because the possibility of attack's spreading from one user to another might differ from attack's spreading in the opposite direction that implies cycle's existence. A* algorithm is convenient because it makes the searching of distance only between two vertices. But selection of the right Heuristic function makes using the algorithm more difficult. Also A* algorithm uses large amount of memory during the work. That is why it is not reachable to use it for solving our problem. Dijkstra's algorithms is suited for it, because it has computational complexity $O(n^2)$. The best complexity for algorithms that are based on Dijkstra's one is $O(n \ln n + m)$ and it is achieved be providing data as Fibonacci heaps. But constants that are hidden in asymptotic estimation of modification's intensity are often too big in practice. One the other hand data can be stored in the binary heap and then the complexity will be $O(n \ln n + m \ln n)$. But let us notice that the time of modification's work will be reduced only if $m = n^2$ (in case of sparse graph). But using sparse graphs is a very rare case in our task.

Thus, taking into account specified limitations, the most appropriate variant is Dijkstra's algorithm which gives not the best time but available to work with any input data. The additional conditions were introduced to get faster work. To be exact, the threshold was set to reduce the computational complexity: if the path from the initial vertex to the processed one becomes less than a given threshold we have to exclude current vertex from consideration. Also, in the path to final vertex is already found than according to our task the algorithm can be finished.

4. Implementation.

The solution described above has been implemented as a new class (CriticalPath) for the "Attack analyzer" module, which was developed in [2]. This module is one of the main modules of a set of programs designed to obtain automated assessments of the security of users of information systems

from multi-way socio-engineering attacks. The main method of the added class is FindCriticalPathByIds. Let us consider it in more detail.

As an input parameter, the method takes the ID of two users of the social network VKontakte and the object SocialGraph. Initially, two HashMap type objects are created: LengthCriticalPath and WantedCriticalPath, the keys of which are the user IDs, and the values are the length of the current shortest path to the given user in the logarithmic form and the user ID from which they switched to this one, respectively. The structure of the HashMap is used in connection with the need for quick access by ID. Initially, in objects such as HashMap, the first user is located. The set VisitedNodes is responsible for the nodes of the graph, the critical path to which is already found, such are the vertices returned by the method FindLowestDistanceUser. The program ends its execution if the given method returns the ID of the second user, or if a zero value is returned, but in this case the path either does not exist or is too unlikely. If these conditions are not met, the SocialGraph class method (getAllConnectionById) searches for all users to which you can directly go from the user found. All such users are checked for belonging to the LengthCriticalPath, and if they don't belong to this object, the possibility of reducing the critical path is checked. The information is then entered in LengthCriticalPath and WantedCriticalPath. Note that at this stage, the path value is also compared with the threshold value, and if the path value is larger, the user will not be added. Figure 1 shows a block diagram of the algorithm.

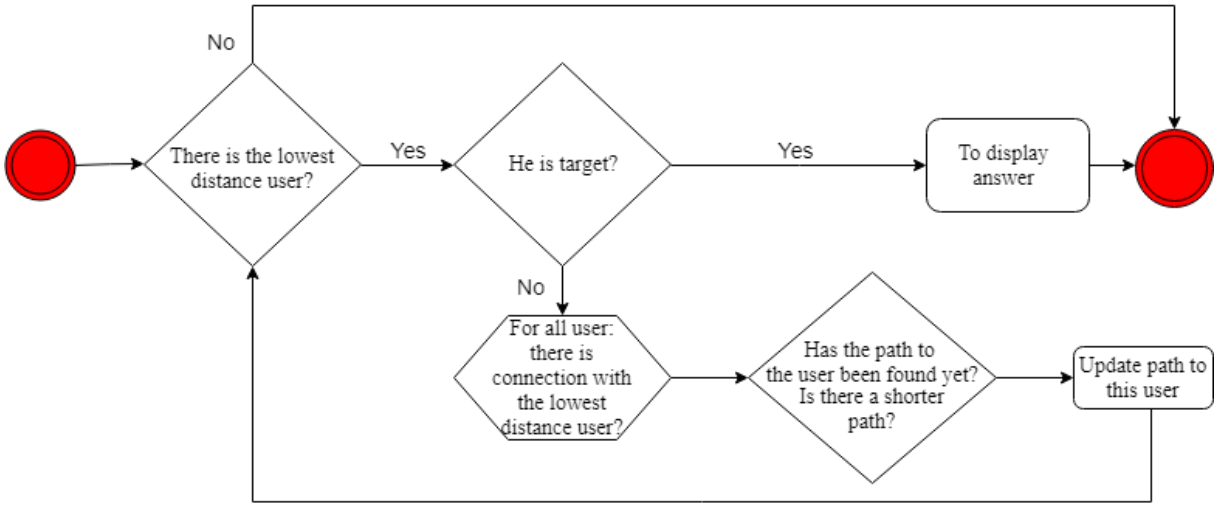

Figure 1. Flowchart of algorithm.

The JGraphX library is used to present the output of the program in visual form. Screenshots of a running program is presented in Figure 2.

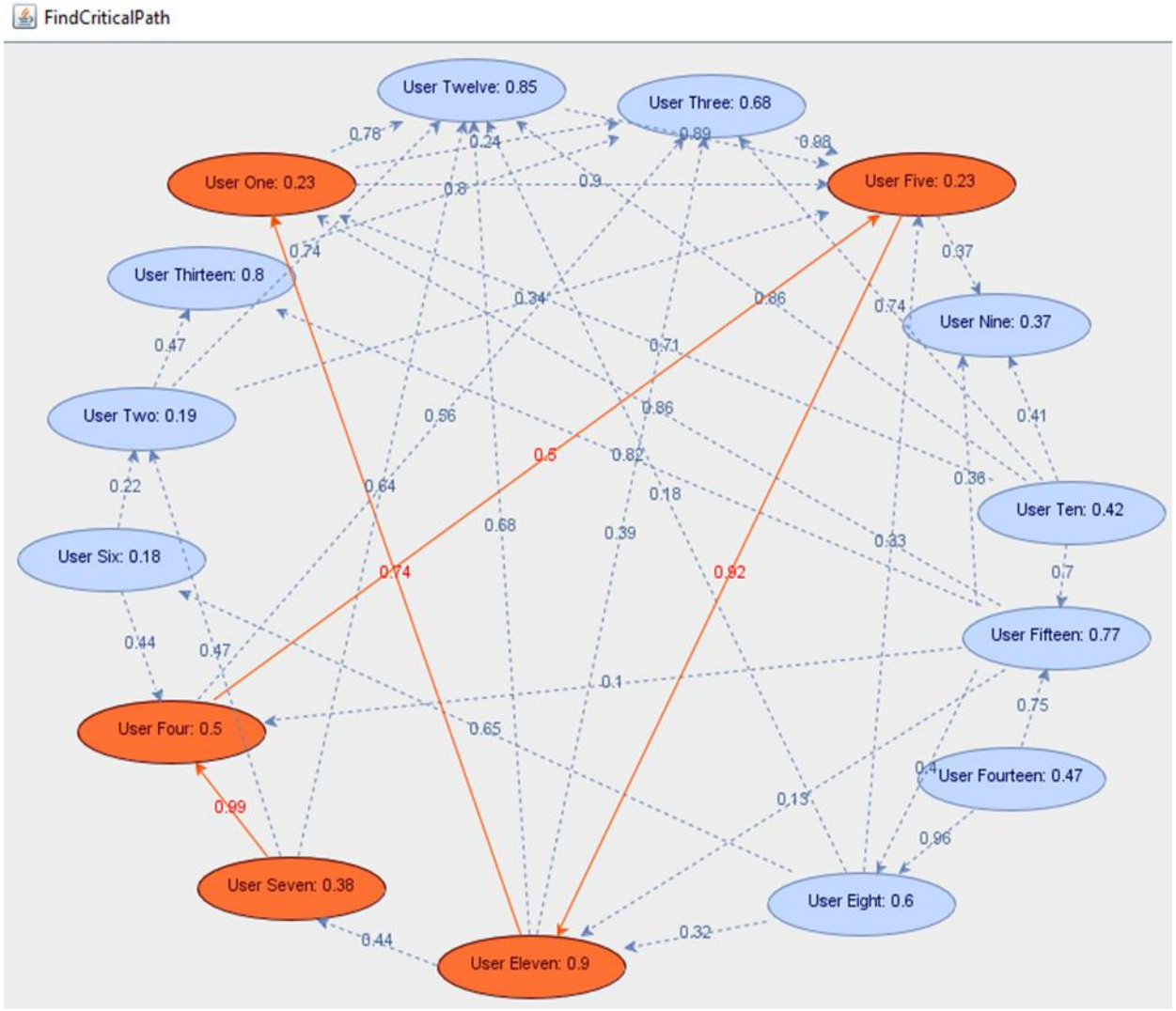

Figure 2. Screenshots of a running program (Critical path searching between User Seven and User One).

5. Conclusion.

The practical significance of the research is to expand the capabilities of the existing software package and further application of the module in examining the security of users of information systems in multi-way socio-engineering attacks. This result became possible due to a mathematical apparatus, which allowed reducing the original problem. Prospects for further research are to find new approaches for modelling and assessing the probabilities of the success of the spread of multi-way socio-engineering attacks, in particular, Bayesian network apparatus [9] can be involved.

6. References

[1] Azarov A A, Tulupyeva T V, Suvorova A V, Tulupyev A L, Abramov M V, Usupov R M 2016 *Sotsioinzhenernye ataki: problemy analiza* (SPb.: Nauka) ed R M Usupov

[2] Abramov M V 2018 *Metody i algoritmy analiza zashchishchennosti pol'zovatelej informatsionnyh sistem ot sotsioinzhenernyh atak: otsenka parametrov modelej* (SPb: SPIIRAS) pp 129-157

[3] Cormen, T H, Leiserson C E, Rivest R L, Stein C 2001 Single-Source Shortest Paths and All-Pairs Shortest Paths. *Introduction to Algorithms* (MIT Press and McGraw-Hill.) 2nd ed. pp 580–642

[3] Levitin A 2012 *Introduction to the design & analysis of algorithms* (USA: Addison-Wesley) 3rd ed pp 304-37
[5] TSB ozhidaet rosta aktivnosti moshennikov, ispol'zuyushchih sotsial'nuyu inzheneriyu. https://ria.ru/economy/20171213/1510861611.html [accessed: 07.05.2018]
[6] Po sledam CyberCrimeCon 2017: Tendentsii i razvitie vysokotekhnologichnoj prestupnosti. https://habr.com/company/group-ib/blog/341812/ [accessed: 16.04.2018]
[7] Abramov M V, Tulupyev A L, Sulejmanov AA 2018 Analysis of users' protection from socio-engineering attacks: social graph creation based on information from social network websites *Scientific and Technical Journal of Information Technologies, Mechanics and Optics* **18** pp 313–321
[8] Rassel S, Norvig P 2009 *Artificial Intelligence: A Modern Approach*. (London: Prentice-Hall International) 3nd ed. pp 93-99
[9] Haritonov N A, Berezin A I Sintez matematicheskogo predstavleniya aciklicheskoj algebraicheskoj bajesovskoj seti *Proc. Int. Conf. on soft computing and measurements* (vol 1-2) (SPb: Saint-Petersburg Electrotechnical University) pp 141–143.

Acknowledgments
The research was carried out in the framework of the project on State assignment No. 0073-2018-0001, with the financial support of the RFBR (project No. 18-37-00323 Social engineering attacks in corporate information systems: approaches, methods and algorithms for identifying the most probable traces; project No. 18-01-00626 Methods of representation, synthesis of truth estimates and machine learning in algebraic Bayesian networks and related knowledge models with uncertainty: the logic-probability approach and graph systems).