Online social network profile data extraction for vulnerability analysis

Sophia Alim*, Ruqayya Abdulrahman, Daniel Neagu and Mick Ridley

AI Research Centre,
University of Bradford,
Bradford BD7 1DP, UK
E-mail: S.Alim@Bradford.ac.uk
E-mail: R.S.H.Abdul-Rahman@Bradford.ac.uk
E-mail: D.Neagu@Bradford.ac.uk
E-mail: M.J.Ridley@Bradford.ac.uk
*Corresponding author

Abstract: The increase in social computing has provided the situation where large amounts of personal information are being posted online. This makes people vulnerable to social engineering attacks because their personal details are readily available. Our automated approach for personal data extraction was developed to extract personal details and top friends from MySpace profiles and place them into a repository. An online social network graph was generated from the repository data where nodes represent peoples' profiles. Analysis was carried out into what factors affect node vulnerability. The graph analysis identified structural features of the nodes, e.g., clustering coefficient, indegree and outdegree, which contribute towards vulnerability. From this, it was found that the number of neighbours and the clustering coefficient were major factors in making a node vulnerable because of the potential to spread personal details around the network. These results provide a good foundation for future work on online vulnerability in online social networks (OSNs).

Keywords: online social network; OSN; vulnerability; information disclosure; automated data retrieval.

Reference to this paper should be made as follows: Alim, S., Abdulrahman, R., Neagu, D. and Ridley, M. (2011) 'Online social network profile data extraction for vulnerability analysis', *Int. J. Internet Technology and Secured Transactions*, Vol. 3, No. 2, pp.194–209.

Biographical notes: Sophia Alim graduated in 2006 with BSc (Hons.) in Business Information Systems from the University of Salford, UK. In 2007, she received her MSc in Computing from the University of Bradford UK. At the same university, currently, she is working towards a PhD with Dr. Daniel Neagu and Mr. Mick Ridley as her supervisors. Her research focuses on the ever evolving area of social networking and how the issue of privacy is going to affect the structure and information disclosure of these networks. Her motivation for her research comes from her desire to reflect the multidisciplinary areas of computing. Her research interests include web accessibility and social networking.

Ruqayya Abdulrahman is a Lecturer in Computer Science at Taibah University, Saudi Arabia. In 2002, she obtained her BSc (Hons.) in Computer Science from King Abdulaziz University in Saudi Arabia. In 2007; she was awarded an MSc with distinction in Software Engineering by the University of Bradford, UK. Currently, she is a PhD student at the School of Computing, Informatics and Media of the University of Bradford. Her research addresses software agents, web database processing, data retrieval, online social network and software engineering.

Daniel Neagu is a Senior Lecturer in Computing at the University of Bradford. His research interests include knowledge discovery, information retrieval, data mining applications in multidisciplinary projects (with a focus in online social networks, healthcare and web profiling) by fusion of human experts knowledge and AI tools. He is a co-author of over 70 peer-reviewed publications and is the Principal Investigator in projects funded by EU FP7, FP5, EPSRC, BBSRC and UK industry. He is an IEEE Technical Expert, ACM and BCS member and a Higher Education Academy Fellow.

Mick Ridley is the Head of the Department of Computing at the University of Bradford. He is a member of the CPHC Learner Development Group. He is co-author of two books on databases. His research interests include databases, web database systems, bibliographic databases, XML and databases. In bibliographic applications this included BOPAC and he was a member of the Joint Steering Committee for Revision of Anglo-American Cataloguing Rules: Format Variation Working Group.

1 Introduction

The popularity of online social networking sites has increased the amount of personal data which is distributed on the net. This is supported by the fact that social networking sites have overtaken e-mail in terms of usage (BBC, 2009a). Online social networking sites contain user profiles which consist of personal data. These profiles are semi-structured as described by Widom (1999) and the profile data or structure may change in an unpredictable way. This fits in with the way online social networks (OSNs) operate. Social network profiles change all the time, not just in structure but in content as well. We have identified the need for more research that has to be carried out into the extraction from semi-structured pages in terms of online social networking profiles.

The motivation for this paper is that as far as the paper authors' know, there has been little research associated with automated extraction methods in regards to semi-structured web pages from OSNs. Our goal is to extract the relevant profile data in order to produce an OSN graph. The graph will be analysed for structural features that contribute towards making nodes vulnerable to social engineering attacks. Each node represents a profile on the OSN and an edge represents the friendship links between the profiles.

Also the extracted data can be mined in the future to find attributes that can cause the profile owner to be vulnerable. In terms of OSNs, our research will allow us in the future to investigate the friends of a profile and see if any of the friends have profiles on other OSNs. This links into the transitivity concept where, e.g., A and B are friends on one OSN, but B and C are also friends on another OSN. The question is: will A and C become friends and if so how great will the strength of their friendship be. The question

posed fits with Granovetter's (1983) theory about weak ties and how they can provide an alternative information source to the ones associated with the strong ties.

For data to be mined, in the first place it has to be extracted. This paper's main focus is on the automated extraction process of personal details from online social networking sites and the analysis of the graph. Our approach will aim to lower the cost of information retrieval because the attributes from the OSN profiles will be extracted and inserted into a local repository. Data analysis can then take place offline. Due to the fact that OSN profiles in general will change in terms of structure and content on a regular basis, a timestamp can be used to help track changes and consequently vulnerability. These changes can be followed over time.

The structure of our paper is as follows. Section 1 introduces the problem statement as well as our research contribution. Section 2 explores related work on data extraction from World Wide Web pages, investigating the issues that surround data extraction and analysing various approaches to combat the data extraction issues. Section 3 presents our data retrieval approach as highlighted in Alim et al. (2009) in detail and the reasoning behind it. Section 4 discusses the findings and the analysis of the graph in regards to vulnerability. Section 5 highlights the conclusions and suggests ideas for future research.

2 Related work

Data extraction is a field that is concerned with grabbing information from different web resources including websites, online databases and services. It is necessary to find tools for data extraction because of the dynamic nature of the World Wide Web. This creates some difficulties for end users and application programs when it comes to finding useful data.

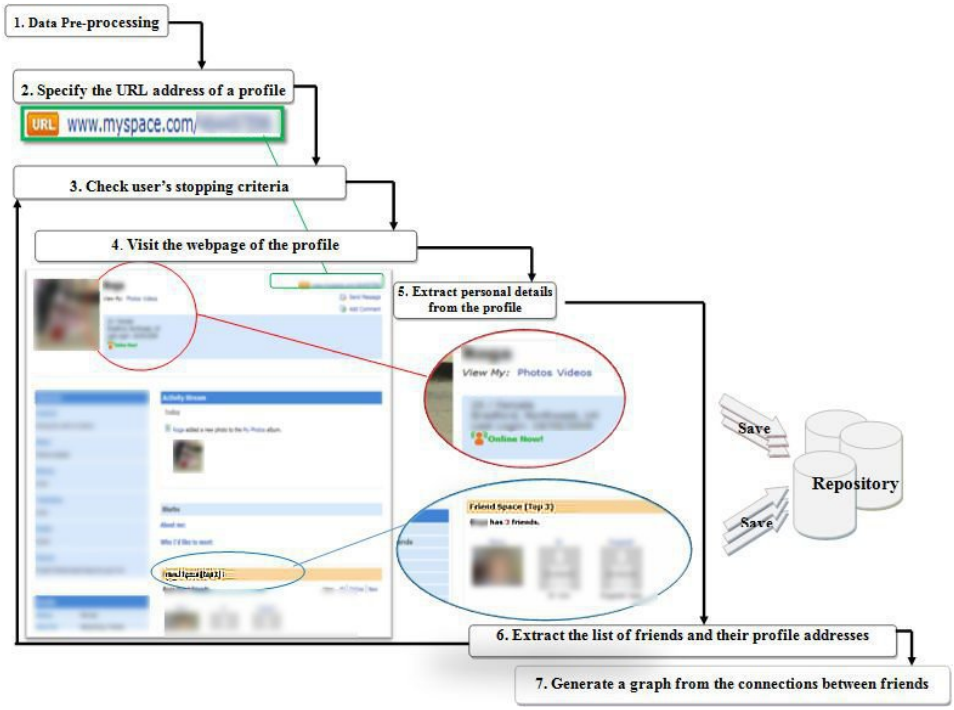
There are several issues which contribute to users and applications failing to find required web pages. One of those issues is related to information representation. Data on web pages can be found in different formats. HTML is designed for unstructured data which contains information in several formats, e.g., text, image, video and audio. It is known that web pages in HTML format are 'dirty' because their contents are ill-formed and 'broken' (Ma et al., 2003). In contrast, XML and XHTML are designed for more structured data. They are stricter in terms of having well-formed documents i.e., the documents' contents should conform to their syntax rules. This feature helps the parsers of search engines to interact with the web pages' contents more efficiently (Ma et al., 2003). One of the useful techniques is wrappers as specified by Palmieri et al. (2004), and Liu and Zhai (2005). Wrappers are responsible for converting HTML documents into semantically meaningful XML files to simplify the operation of extracting data. Wrappers are not efficient though because the programmers have to find the reference point and the absolute tag path of the targeted data content manually. This requires one wrapper for each website since different sites follow different templates. The effects are increased time consumption and effort from the programmer.

Another issue that prevents efficient data extraction is related to the technique used by search engines to find related web pages. Search engines depend on crawlers to search the World Wide Web for the required keyword(s) that are entered by end users. Crawlers collect information about the visited website then record this information in a process called indexing which is used later for ranking websites. A problem has arisen because of the limitation of crawlers' capabilities. Crawlers can cover only the publicly indexable

web (PIW) while the majority of useful data is in the hidden or deep web, which is not reachable by crawlers as highlighted by Lawrence and Giles (1998), and Bergman (2000). Deep web pages are described by Park and Barbosa (2007) as dynamic pages listing data from databases using a predefined format. Their content is likely to be of very high quality since they are managed by organisations interested in maintaining accurate and useful databases.

Extracting data from deep web pages has previously been approached in Park and Barbosa (2007) to deal with drawbacks of some other work tailored to specific websites. In Zhang and Shasha (1997), the approach did not work so well on loosely structured records because they depend on a tree-edit distance metric. The suggested method by Park and Barbosa (2007) avoids those weaknesses by using the web data extractor algorithm which depends on clustering and the weighted tree matching metric to extract data. Liu and Zhai (2005) realised the importance of extracting data records that were retrieved from databases and displayed on web pages. They analysed the disadvantage of the approaches that were used for extracting data i.e., wrapper induction and automatic extraction, then they proposed a method called nested data extraction using tree matching and visual cues (NET) for extracting flat or nested data records automatically.

Figure 1 The approach for automated data extraction (see online version for colours)



Since a large amount of information is stored in web databases that are hidden and not indexed by search engines, Hedley et al. (2004) generated a method that will detect the templates then analyse the textual content and the document's adjacent tag structure to extract query related data. Crescenzi et al. (2004) demonstrated a system to grab data from the web automatically. The proposed system is also similar to two other approaches:

in the first one, Kao et al. (2004) are concerned with analysing news websites by identifying pages of news indexes and pages containing news; their work aims to classify pages according to their structure, without any previous assumption. Chakrabarti et al.'s (1999) approach is focused on crawling: in contrast to Kao et al. (2004), this system is focused on structure recognition rather than searching for pages which are relevant to the topic. This approach does not crawl data behind forms.

Our approach, detailed in Section 3, is similar to the work of Park and Barbosa (2007), but the difference is that we concentrate more on the structure of the profile and the corresponding tokens. In social networking, users can use their imagination when filling in personal details. In Park and Barbosa (2007), using patterns to extract profile data can be strict and therefore, it may miss out data that does not fit the pattern. Our data retrieval approach which is detailed in Alim et al. (2009) and illustrated in Figure 1, proposed extracting attributes and a list of top friends from a MySpace profile. MySpace was chosen because it allows a rich source of data to be derived from profiles without the need to be a member of MySpace. Even if the profile is private, we can still derive some attributes as highlighted in Figure 1.

3 Data retrieval approach

Automated data extraction is just starting to be used in research regarding online social networking (Caverlee and Webb, 2008).

Table 1 Different extraction methods in regards to OSNs

Method	*Research study*	*Ref*
Developed an automated web crawler using the Ruby programming language. The crawler would visit profile pages based on a randomly generated list of id numbers using the RAND function of Microsoft excel. Regular expressions were used to collect the relevant bits of data.	Age differences in online social networking	Arjan et al. (2008)
Wrote two crawlers that were MySpace specific based on 'Perl's LWP user agent and HTML parser modules'. They gathered two datasets. One was collected using random sampling and the other one with relationship-based sampling.	A large scale study of MySpace observations and implications for online social networks	Caverlee and Webb (2008)
Downloaded MySpace profiles randomly.	Social networks, gender and friending: an analysis of MySpace member profile	Thelwall (2008)
Used a random number generator to decide which profiles to analyse. Analysis of the profiles took place by manually analysing the profiles and using a data collection form to record their findings	Personal information of adolescents on the internet. A quantitative content analysis of MySpace	Hinduja and Patchin (2008)

Table 1 illustrates some of the data extraction techniques used to extract attributes from online social networking profiles. It shows some data extraction methods ranging from manual through to automated methods. The data retrieval process (illustrated in Figure 1) outlines our approach to extracting attributes and a list of top friends from a MySpace

profile. MySpace was chosen because when we tried it as an external user to extract profiles from different OSN, e.g., Facebook and Friendster, either no data or minimal data was made available for public access in the first instance.

3.1 Approach components

Our process comprises of the following components:

Step 1 *Data pre-processing* involves analysis of the HTML structure of a given profile. The HTML content is parsed and a vector of tokens is produced. The extracted tokens help in the design of the tables in the repository. The reason for data pre-processing is because MySpace profiles have different structures and therefore different tokens. We created our own MySpace profiles to help investigate different possible structures and attributes.

Step 2 *Specify the URL address of a profile*. All social network profiles come with a unique profile URL address. The algorithm for extraction of the personal details involved developing and expanding the library which was provided by Haines (1999). This code was developed to be applied to online social networking profiles. We use the URL of the OSN profile as a parameter. Then Java IO methods would be used to extract the HTML of the webpage and store it as a character array. The parsePage method which we defined would remove all the HTML tags from the string, split the remaining text in tokens and place the tokens into a vector. This method proved the most important when extracting the personal details and the list of top friends from the profile. In the case of Figure 1, the URL address, personal details and list of top friends have been blurred out for privacy reasons.

Step 3 *Check the user's stopping criteria*. The users can specify whether they want to stop the extraction by the number of friends extracted (e.g., the first 100 friends) or by the level (e.g., level 1 which is just the top friends of the specified profile) extracted (see Section 3.2).

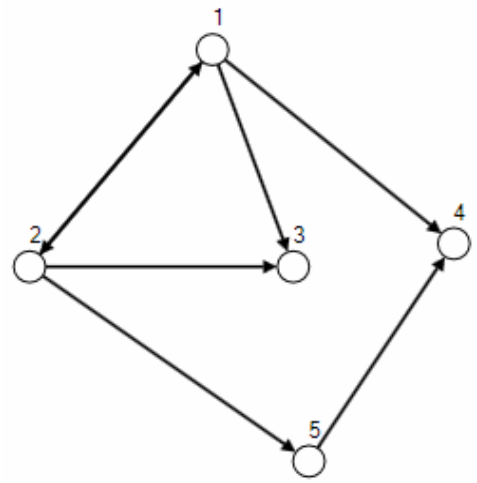
Step 4 *Visit the specified profile webpage* after checking that it has not been visited before. Breadth First Search has been used for our applications to travel the social network as explained below.

Step 5 *Extract the relevant personal details from the profile* and insert them into the repository. The repository that we used was PostgreSQL 8.1.4. The repository structure has to be designed for the Breadth First Search algorithm.

Step 6 *Extract list of friends and their profile addresses* then insert them into the repository if they have not been stored before. In the case of this research paper, the extracted friends' lists just consist of the top friends who we assumed the user may have a strong affiliation with. The data in the repository can be used for data mining purposes in the future to find patterns.

Step 7 *Automatically generates an OSN graph*. The various structural features of the graph will be analysed to see how they contribute towards the vulnerability of a node.

Figure 2 A graph to illustrate Breadth First Search



Breadth First Search was used to travel across the network. The scenario illustrating Breadth First Search (see Figure 2) shows that profiles 2, 3 and 4 are the three friends of profile 1. Profile 2 also has three friends which are profiles 1, 3 and 5. Profiles 1 and 2 are the friends of profile 3 where as profiles 1 and 5 are the friends of profile 4. Entrance to the repository is implemented as a queue system. The arrows that the relationship represents are the 'is a friend of' relationship. An example is that node 3 is a friend of node 1. The bidirectional arrows show that the relationship applies both ways. An example is that node 2 is a friend of node 1 and node 1 is a friend of node 2.

Using Breadth First Search in this case follows the following steps:

1 Add profile 1's attributes and top friends list into the front of the queue ready to go into the repository.

2 Loop:
   a Look at profile 1's friends and check to see if they already exist in the repository. In the first iteration, the friends are profiles 2, 3 and 4.
   b If the friends do not exist in the repository, add their attributes and list of top friends to the rear of the queue.
   c Look at the next profile at the front of the queue (in this case profile 2).
   d Repeat steps a and b.

3.2 Stopping criteria

In regards to stopping extraction, we will allow the user to specify the values of two criteria that we set. The criteria are:

- the number of top friends
- the level of iteration, e.g., a friend of a friend.

In our experiment, we stopped the extraction by the level of iteration which in this case was three because it would give us a large enough sample of profiles to analyse.

3.3 Limitations of current approach

The limitations of the current approach will provide ideas on what to improve in future research. The limitations include:

1 An incomplete list of friends. The list of friends extracted from the profiles at the moment is not the full friends list. It is just the top friends. A full list of friends will give us a more accurate picture about the environment of the profiles.

2 One mode of travel across the graph. Only Breadth First Search was used to travel across the OSN. This may not be the most efficient algorithm to use so Depth First Search needs to be implemented as well so we can compare the performance of the two algorithms in this scenario.

3 Various profile structures. Musicians, magazines and band profiles are not extracted because their profiles were of a different structure. This structural variation made the profiles hard to extract from. Also the friendship between a person and a band differs from that of two individuals who are friends. The friendship between a person and a band is a 'fan-based' relationship compared to a relationship between two individuals which is a 'friend of' relationship. A 'fan-based' relationship is less likely to show characteristics of being transitive or reflexive.

4 The processing of the graph. Graph software can struggle with a large number of nodes. If the number of nodes is up to 10,000 then the processing is fine. 10,000 nodes are classed as a big network. This poses a problem if we ever want to crawl a large amount of data i.e., more than 10,000 nodes and produce an OSN graph. This issue also affects social network metric application using software, e.g., finding the average shortest path of the network. This metric is very computer intensive. We need to develop solutions to the problems associated with the processing of large graphs.

4 Findings and discussion of results

Our findings are based on two areas which include the extraction process and the processing and analysis of the OSN graph. To test the data retrieval process, we started from a randomly selected node and ran the algorithm 500 times. This resulted in personal details and a list of friends being extracted from 298 profiles because the rest of the profiles were private, musicians or bands.

4.1 Extraction findings

From our experimental work (Alim et al., 2009) we have learnt how to automatically extract data from an online social networking profile using a Breadth First Search approach. The structure of MySpace profiles was found to differ depending on the type of profile and the users' preferences. This proved a challenge when implementing the code especially when over time, the developers of MySpace have the ability to change the structure of the profile and therefore change the HTML structure. We identify this as a problem for data extraction from social networks.

Analysis of web structures of various OSN profiles revealed that there was a standard format. Even though some of the profiles were private profiles we could still extract some attributes, e.g., nickname, gender, age and location. Data that is placed in the repository can be mined and analysed offline to recognise patterns and trends about the social network in which the profiles are based in.

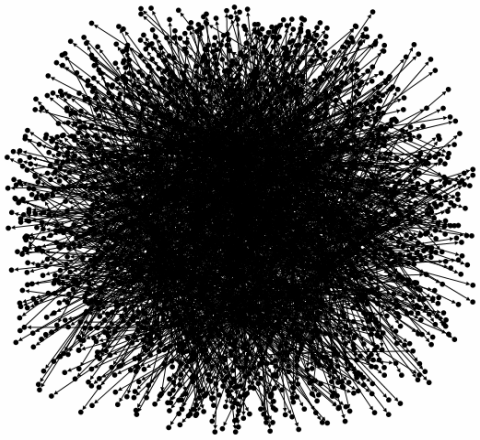
The profile data can also be used to identify which profile attributes and values make the person vulnerable to social engineering attacks. The meaning of vulnerability is associated with the disclosure of personal details. The more details you disclose, the more vulnerable you make yourself.

Vulnerability can be inferred by the attributes presented, e.g., if the age and horoscope signs are present on a profile then it is possible to guess when the birthday is. If there are comments present on the profile as well you may be able to tell the exact date of the birthday. Other attributes that may contribute to a profile being vulnerable include whether they are a drinker or a smoker. Personal details present on OSN profiles can be of particular interest to employers when it comes to hiring employees. Profile details that can cause concern includes any mention of alcohol or drug use and information that implies that the person has been linked to criminal activity (Havenstein, 2008).

4.2 Online social network graph

An OSN graph was then generated from the repository data. The repository data included the personal details and the friends' lists of all the OSN profiles that were crawled using the process outlined in Section 3.

Figure 3 Online social network graph



The graph was generated so we could analyse the graph for characteristics which could influence vulnerability via the spread of personal details in OSN profiles. The graph is illustrated in Figure 3.

The OSN graph is modelled as a directed multigraph $G = (V, E)$. V is the set of nodes which represents the profiles of people on the OSN which has been extracted from. E is the set of edges which links the profiles together. The relationship which is represented by the edges is a top friend relationship. Top friends are not reciprocal, e.g., node 1 can be the top friend of node 2 but node 2 may not be on node 1's top friends list.

The graph G is a directed multigraph because we want to analyse the flow of information so direction is required. Also the multigraph aspect gives an extra dimension to the analysis stage especially with the information flow going bidirectionally. A multigraph allows parallel edges between the nodes and this is modelled with the function: $f : E \to \{\{u, v\} : u, v \in V \text{ and } u \neq v\}$. This function shows that nodes are not connected to themselves. Graph G, for example, has $| V |$ = 2,197 and $| E |$ = 2,747.

4.2.1 Graph characteristics

When analysing OSN graphs, there are many characteristics that can be examined. In terms of emphasising vulnerable nodes, there is a set of structural properties which can help in the analysis. Vulnerability involves both the state of the network and the state of the node. Looking at the average clustering coefficient and the average path length values will give an idea about the state of the network. This is the environment the set of nodes are based in. Also, the concepts introduced below will help in explaining how the graph structure and vulnerability metric link together.

Clustering coefficient of a node (Watts and Strogatz, 1998) reflects how well connected the node's neighbourhood is. In terms of equations used for clustering coefficient, there are two different equations which correspond to directed graphs and undirected graphs. Since directed graphs are used in this study, the clustering coefficient of node n using Watts and Strogatz (1998) equation will be:

$$C_n = \frac{e_n}{\left(k_n \left(k_n - 1\right)\right)} \tag{1}$$

where e_n is the number of edges that exist between the neighbours of node n and k_n is the number of neighbours of node n.

If the value of the clustering coefficient is heading towards 1, then most of the neighbours of a node are connected to each other. On the other hand, if the coefficient value is near 0 then the neighbours are not connected to each other at all.

Examining the average clustering coefficient for all the nodes in the network, calculated using Watts and Strogatz (1998) metric in equation (2), can tell us how well connected or not the nodes are to each other:

$$\overline{C} = \frac{1}{n} \sum_{i=1}^{n} C_i \tag{2}$$

where n is the number of nodes and C_i is the clustering coefficient for each node. In the case of the graph in Figure 3, the average clustering coefficient is 0.031 which shows that the nodes are not well connected in this network.

Another important feature is the average path length of a network, which will reflect how good or bad the information flow is (Watts and Strogatz, 1998). Average path length for graph G is worked out by averaging all the shortest (geodesic) distances between node pairs in the network. Let graph G have a set of nodes V. The notation for the shortest distance between two nodes is $d(v_1, v_2)$ where v_1 and $v_2 \in V$.

The equation for the average path length of graph G would be:

$$p_G = \frac{1}{n*(n-1)} \sum_{a,b} d(v_a, v_b) \tag{3}$$

where n is the number of nodes in the network and $d(v_a, v_b)$ is the shortest distance between two nodes.

The higher the average path length of a network, the harder it is for information to flow across the network. For a more efficient network, a short path length is a good quality to have. In the case of the graph in Figure 3, the average path length is 6.837. Even though this value seems quite high, certain studies have shown that the average path length for social networks can be six or above: Milgram's (1976) experiments stated the value was six but in Leskove and Horvitz's (2007) experiments it was 6.6.

The difference between the two studies is that Milgram's (1976) study involved randomly mailing people with a folder. The folder contained the name and location of the target person they had to send the folder to. The sender may not know the target person, so the sender had to post the folder to an acquaintance that may know the target person. An example of this is that a person from Nebraska was asked to mail the folder to a person in Boston. If the sender did not know the person in Boston they had to send the folder to an acquaintance who may know the target person. On the other hand, Leskovec and Horvitz (2008) study involved analysing 30 billion MSN messenger conversations amongst 240 million people. A communication graph was produced from the data gathered. The graph contained 180 million nodes and 1.3 billion undirected edges.

The structural features of the communication graph, e.g., clustering, diameter of the graph and average path length were analysed and it was found that the average path length of the graph was 6.6. This value of 6.6 indicated that there was some truth in Milgram's (1976) idea involving the six degrees of separation.

In regards to our OSN graph, an average path length of 6.837 shows that the network has a wide diameter and the information will take a while to travel around the network.

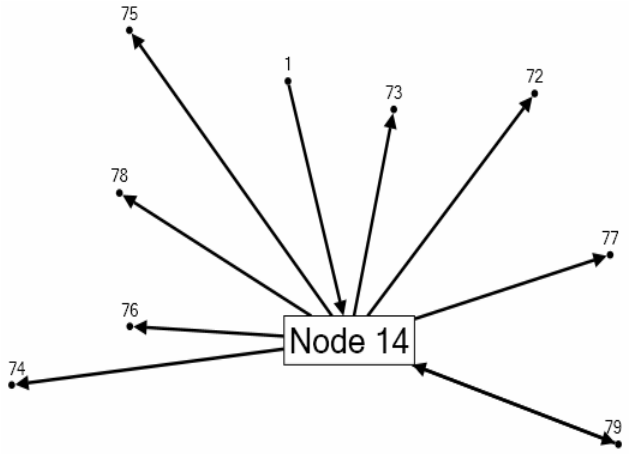
4.2.2 Graph characteristics for vulnerability

Analysing the node as an individual entity in terms of structure is important when talking about vulnerability. This analysis gives us more information about the state of the node and the immediate neighbourhood. The three main characteristics to explore include the indegree, outdegree and clustering coefficient.

The indegree of node n is the number of edges coming towards node n. The indegree of node n is denoted by i_n. The outdegree of node n is the number of edges going away from node n. The outdegree of node n is denoted by on. Figure 4 shows the subgraph of a node and illustrates the indegree and outdegree concept. The node represented by a diamond has an indegree of 2 and an outdegree of 8.

An indegree of node $n(i_n)$ in this case specifies that this profile is a friend of someone and is in his/her top friends list. The number of indegree edges shows how much other people have an interest in that node. If there are many indegree edges, the node is highly interesting and this leads to increased communication regarding that node. This provides a good platform for personal details to spread. If the node is a top friend, important personal details may be exchanged more readily.

Figure 4 Indegree and outdegree for a node in an OSN



An outdegree (o_n) represents how many friends the node has in his/her top friends list. If a node has many outdegree edges this shows that the node has many friends they can pass information onto. Also the node can make itself vulnerable to attack from its own friends.

The most important factor in regards to the vulnerability is the number of neighbours the node has, which can be partially worked out by calculating the total degree.

With multigraphs allowing parallel edges you have to be careful that the neighbours are not counted twice. In our experiment we have taken this issue into account. The total degree of node n is illustrated in equation (4).

$$D_n = i_n + o_n \tag{4}$$

A higher degree total indicates that the node has many neighbours therefore making the node more vulnerable. Also, a high outdegree value for the node indicates that there is scope for the information to spread further into the network and to other sub networks.

Another feature that plays an integral part in accessing the vulnerability of a node is its clustering coefficient. A node with a high clustering coefficient will have a neighbourhood where information will flow easily due to nodes knowing each other. In terms of vulnerability, a node with a clique where all the nodes in the neighbourhood know each other will have a greater vulnerability value.

An interesting aspect that has been highlighted in the media by BBC (2009b) is the issue of the average number of friends a node can have. This issue stems from Dunbar's number which is 150. The number originates from Dunbar's (1992) work into how many humans a person can maintain a stable relationship with. In our research study, the average number of top friends a node has is 2.3. Bearing in mind that Dunbar's number is associated with all friends; a possible explanation for the low number of top friends is that most people may not have really close friends. The top friend relationship outlines that a strong connection exists between the node and its friend. It may be difficult for the person to categorise their top friends. The graph in Figure 5 illustrates the fluctuations in the number of top friends.

Figure 5 Top friends distribution

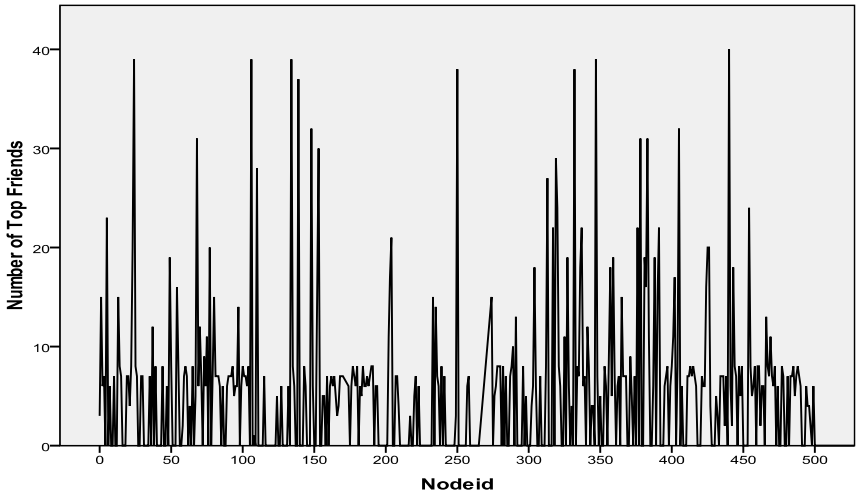


Figure 5 shows that there are nodes that have a high number of top friends, e.g., 40. This can indicate a dangerous situation because of the notion of competition between the nodes. If a node has 40 friends there is the possibility that some of the top friends are in fact relative strangers. This can lead to increased vulnerability for the node because they are trusting people they do not know with their personal details. These are the sort of personal details which are used a lot as authentication into systems or identifiers, e.g., pet's name as a 'security question'.

An insight into graph analysis has allowed us to identify factors which affect the vulnerability of a node. These factors can be taken into account when working out vulnerability empirically.

5 Conclusions and future research

With the increase of popularity in OSNs, more personal details are being placed online. This can lead to making people vulnerable to social engineering attacks. The two components which need to be analysed to determine the level of vulnerability include the extracted personal data and the structure of the graph representing the relationships between the nodes. The graph can be analysed in terms of the network state and the node state. Our experiment has shown that analysing the indegree, outdegree and clustering coefficient of a node tells us about the vulnerability of the node. These features are factors that contribute towards vulnerability.

A node that has a high indegree, outdegree and clustering coefficient is most likely to be vulnerable. This is due to the node having a high number of neighbours which can spread the personal data of the node into their sub networks. If the node has a high indegree then this node is trusted by other nodes. On the other hand, if the node has a high outdegree then they have the ability to spread the information across the network.

Also a high clustering coefficient means the immediate neighbours of the node are all connected to each other. This means that the personal details will flow faster. This is

dependent on the strength of the relationship between the nodes. More research will have to be done into that aspect.

In terms of data extraction, our research has shown how far social network extraction has come since the days when extracting attributes involved a lot of interaction with the profile owner, e.g., questionnaires and interviews. Automatic extraction of attributes is the way forward because it allows us to process much larger volumes of data which can be extracted over a period of time. Also, automated extraction is independent of user involvement and it can happen with semi-structured web pages. The main challenge when carrying out the experiment to implement the automated data retrieval approach was that social networks like MySpace have more than one profile structure template and the user can customise the template.

This research has provided opportunities for future research to be carried out, as listed below:

1 Extract all friends from online social networking profiles instead of just top friends. Then analyse the graph and see how the vulnerability of nodes is affected.

2 Run the application over a period of time to track the changes in vulnerability of the OSN and acknowledge the timestamp of current repository content.

3 Development of an agent to automate the process of data retrieval. The agent will be able to track the behaviour of the profile and report any changes.

4 Extracting the data from the online social networking profiles using a Depth First Search. The results from this approach can then be compared to the results from Breadth First Search. The speed and the amount of memory used will be analysed to see which searching algorithm would be the most productive in terms of this application.

5 Extract from other OSNs users with registered profiles. Extraction from different networks will generate OSN graphs with different behaviours. Vulnerability of nodes will change due the behaviour of the profiles in the social network.

Our study into the extraction and vulnerability analysis of OSN profiles has given us an opportunity to explore a new field in social network analysis.

References

Alim, S., Abdul-Rahman, R., Neagu, D. and Ridley, M. (2009) 'Data retrieval from online social networking profiles for social engineering applications', *Procs. of the 4th International Conference for Internet Technology and Secured Transactions ICITST-2009, Springer Lecture Notes in Computer Science*, pp.207–211, London, UK.

Arjan, R., Pfeil, U. and Zaphiris, P (2008) 'Age difference in online social networking', *Extended Abstracts on Human Factors in Computer Systems CHI'08*, ACM, pp.2739–2744, Florence, Italy.

BBC (2009a) 'Social sites eclipse e-mail use', available at http://news.bbc.co.uk/1/hi/technology/7932515.stm (accessed on 4 April).

BBC (2009b) 'What's the ideal number of friends?', available at http://news.bbc.co.uk/1/hi/7920434.stm (accessed on 14 April).

Bergman, M.K. (2000) 'The deep web surfacing hidden values', available at http://grids.ucs.indiana.edu/courses/xinformatics/searchindik/deepwebwhitepaper.pdf (accessed on 23 April 2009).

Caverlee, J. and Webb, S. (2008) 'A large-scale study of MySpace: observations and implications for online social networks', *Procs. of the 2nd International Conference on Weblogs and Social Media (ICWSM 2008)*, pp.36–44, Seattle, USA.

Chakrabarti S., Van den Berg, M. and Dom, B. (1999) 'Focused crawling: a new approach to topic specific web resource discovery', *Computer Networks*, Vol. 31, No. 11, pp.1623–1640.

Crescenzi, V., Mecca, G., Merialdo, P. and Missier, P. (2004) 'An automatic data grabber for large web sites', *Procs. of the International Conference on very Large Data bases (VLDB 2004)*, pp.1321–1324, Toronto, Canada.

Dunbar, R.I.M. (1992) 'Neocortex size as a constraint on group size in primates', *Journal of Human Evolution*, Vol. 22, pp.469–493.

Granovetter, M.S. (1983) 'The strength of the weak tie: revisited', *Sociological Theory*, Vol. 1, No. 1983, pp.201–233.

Haines, S. (1999) *Java 2 from Scratch*, QUE, Canada.

Havenstein, H. (2008) 'One in five employers uses social networks in hiring process', available at http://www.computerworld.com/s/article/9114560/One_in_five_employers_uses_social_networks_in_hiring_process (accessed on 11 March 2010).

Hedley, Y.L., Younas, M., James, A. and Sanderson, M. (2004) 'Query related data extraction of hidden web documents', available at http://dis.shef.ac.uk/mark/publications/my_papers/SIGIR2004HedleyYounasJamesSanderson.pdf (accessed on 4 April 2009).

Hinduja, S. and Patchin, J.W. (2008) 'Personal information of adolescents on the internet: a quantitative content analysis of MySpace', *Journal of Adolescence*, Vol. 31, No. 1, pp.125–146.

Kao, H., Lin, S. and Chen, M. (2004) 'Mining web informative structures and contents based on entropy analysis', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No. 1, pp.41–55.

Lawrence, S. and Giles, C.L. (1998) 'Searching the World Wide Web', *Science*, Vol. 280, No. 5360, pp.98–100.

Leskovec, J. and Horvitz, E. (2008) 'Planetary-scale views on a large instant-messaging network', *Procs. of the 17th International Conference on the World Wide Web (WWW 2008)*, Beijing, China, pp.21–25.

Liu, B. and Zhai, Y. (2005) 'NET – a system for extracting web data from flat and nested data records', *Procs. of 6th International Conference on Web Information Systems Engineering (WISE 05)*, pp.487–495, New York, USA.

Ma, L., Goharian, G. and Chowdhury, A. (2003) 'Automatic data extraction template generated web pages', *Procs. of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 03)*, pp.642–648, Las Vegas, Nevada, USA.

Milgram, S. (1976) 'The small world problem', *Psychology Today*, Vol. 1, No. 1, pp.60–67.

Palmieri, J., Altigram, L., Da Silva, S., Golgher, P.B. and Laender, A.H.F. (2004) 'Automatic generation of agents for collecting hidden web pages for data extraction', *Data and Knowledge Engineering*, Vol. 49, No. 2, pp.177–196.

Park, J. and Barbosa, D. (2007) 'Adaptive record extraction from web pages', *Procs. of the 16th International Conference of the World Wide Web (WWW 2007)*, ACM, pp.1335–1336, Banff, Alberta, Canada.

Thelwall, M. (2008) 'Social networks, gender and friending: an analysis of MySpace member profiles', *Journal of the American Society for Information Science and Technology*, Vol. 59, No. 8, pp.1321–1330.

Watts, D.J. and Strogatz, S.H. (1998) 'Collective dynamics of small world networks', *Nature*, Vol. 393, No. 6684, pp.409–410.

Widom, J. (1999) 'Data management for XML: research directions', *IEEE Data Engineering Bulletin, Special Issue on XML*, Vol. 22, No. 3, pp.44–52.

Zhang, K. and Shasha, D. (1997) *Tree Pattern Matching*, ACM, pp.341–371, Oxford University Press, Oxford, UK.