Andrew Blyth and Iain Sutherland (Eds)

EC2ND 2006

Proceedings of the Second European Conference on Computer Network Defence, in conjunction with the First Workshop on Digital Forensics and Incident Analysis

Faculty of Advanced Technology,
University of Glamorgan, Wales, UK

Springer

EC2ND 2006

Andrew Blyth and Iain Sutherland (Eds)

EC2ND 2006

Proceedings of the Second European Conference on Computer Network Defence, in conjunction with the First Workshop on Digital Forensics and Incident Analysis

Faculty of Advanced Technology, University of Glamorgan, Wales, UK

Springer

Andrew Blyth, BSc, MSc, PhD
Iain Sutherland, BSc, MSc, PhD
School of Computing, University of Glamorgan, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN-10: 1-84628-749-9 Printed on acid-free paper
ISBN-13: 978-1-84628-749-7

Printed in the United Kingdom (ATH)

9 8 7 6 5 4 3 2 1

Springer Science+Business Media
springer.com

Dear Delegates

It is with great pleasure that we welcome you to the second European Conference on Computer Network Defence (EC^2ND) hosted by the Faculty of Advanced Technology at the University of Glamorgan. This year the inaugural Workshop on Digital Forensics and Incident Analysis (WDFIA 2006) is being held in conjunction with the main conference and these proceedings also incorporate the works of WDFIA.

The call for papers has, as in the previous year, attracted submissions from around the globe. This volume contains a selection of double peer reviewed papers from China, Germany, Hungary, Italy, Saudi Arabia, Sweden, Turkey and the U.K. Once more we truly hope that you will enjoy your time here and that the presentations of the contributing authors will provide the grounds for discussion and thought.

We would like to thank the members of the Programme Committees of both events for kindly contributing with their time in the review process. We would also like to thank the Faculty of Advanced Technology for hosting us and our colleagues here at Glamorgan, in particular Caroline Bowen and Theodore Tryfonas who strived to arrange everything in detail and without whom this conference could not have taken place.

And lastly, but no least, we would like to thank you, the authors of papers and the delegates present, as these events would not have been possible without your input and contributions.

Andrew Blyth
Second EC^2ND Chair

Iain Sutherland
First WDFIA Chair

Second EC²ND & First WDFIA Chairs
Andrew Blyth & Iain Sutherland, University of Glamorgan, UK

Joint Programme Committees
Bob Askwith, Liverpool John Moores University, UK

Susan Brenner, University of Dayton, USA

Phil Brooke, University of Teeside, UK

Bill J Buchanan, Napier University, UK

Tim Charrot, QinetiQ, UK

Paul Chichester, CESG/GCHQ, UK

Nathan Clark, University of Plymouth, UK

Tim Cossali-Francis, DSTL, UK

Jon Evans, Gwent Police Hi-Tech Crime Unit, UK

Steve Furnell, University of Plymouth, UK

Stefanos Gritzalis, University of the Aegean, Greece

Alastair Irons, University of Northumbria, UK

Hamid Jahankhani, University of East London, UK

Andy Jones, British Telecom, UK

George Kalb, John Hopkins University, USA

Maria Karyda, University of the Aegean, Greece

Socratis Katsikas, University of the Aegean, Greece

Evangelos Markatos, University of Crete, Greece

Stephen Mason, St Pauls Chambers, UK

Madjid Merabta, Liverpool John Moores University, UK

Lilian Mitrou, University of the Aegean, Greece

David Parish, Loughborough University, UK

Ahmed Patel, Kingston University, UK

Paula Thomas, University of Glamorgan, UK

Panagiotis Trimintzios, ENISA, EU

Theodore Tryfonas, University of Glamorgan, UK

Theodora Tsikrika, Queen Mary University of London, UK

Craig Valli, Edith Cowen University, Australia

Matt Warren, Deakin University, Australia

Local Organisers
Faculty of Advanced Technology, University of Glamorgan, Wales, UK

Caroline Bowen

Theodore Tryfonas

Vivienne Mee

Olga Angelopoulou

TABLE OF CONTENTS

The Second European Conference on Computer Network Defence, inc. the First Annual Workshop on Digital Forensics & Incident Analysis

Section I: Computer Network Defence

Efficient Sampling of the Structure of Crypto Generators' State Transition Graphs

Jörg Keller

FernUniversität, LG Parallelität und VLSI, 58084 Hagen, Germany
Joerg.Keller@fernuni-hagen.de

Abstract. Cryptographic generators, e.g. stream cipher generators like the A5/1 used in GSM networks or pseudo-random number generators, are widely used in cryptographic network protocols. Basically, they are finite state machines with deterministic transition functions. Their state transition graphs typically cannot be analyzed analytically, nor can they be explored completely because of their size which typically is at least $n = 2^{64}$. Yet, their structure, i.e. number and sizes of weakly connected components, is of interest because a structure deviating significantly from expected values for random graphs may form a distinguishing attack that indicates a weakness or backdoor. By sampling, one randomly chooses k nodes, derives their distribution onto connected components by graph exploration, and extrapolates these results to the complete graph. In known algorithms, the computational cost to determine the component for one randomly chosen node is up to $O(\sqrt{n})$, which severely restricts the sample size k. We present an algorithm where the computational cost to find the connected component for one randomly chosen node is $O(1)$, so that a much larger sample size k can be analyzed in a given time. We report on the performance of a prototype implementation, and about preliminary analysis for several generators.

1 Introduction

Stream cipher generators, like the A5/1 in cellular telephones [1, 2] and pseudo-random number generators, are widely used in cryptographic communication protocols. Basically, they are finite state machines that are initialized into a state and then assume a sequence of states completely determined by their transition function $f : N \to N$, where N is their state space, i.e. a set $N = \{0, \ldots, n-1\}$. For given N and f, one can define the state transition graph $G_f = (V = N, E = \{(x, f(x)) : x \in N\}$. Such a directed graph, where each node has exactly one outgoing edge, has a number of weakly connected components (WCC), each consisting of one cycle and a number of trees directed towards their roots, where the roots sit on the cycle. One is interested in the number of the WCCs, their sizes and cycle lengths. The cycle length represents the generator's period, the component size the fraction of nodes that, when chosen as initial state, lead to a certain period. If the period length is too small, then this may hint towards predictability. Furthermore, as a cipher generator shall, in some sense,

randomize, its graph should look randomly as well. If its structure deviates significantly from expected values for random graphs with outdegree 1, this may hint towards a weakness or a backdoor. In this sense, the computation of such a graph's structure can be considered as a distinguishing attack.

Unfortunately, the period lengths of such generators cannot be derived analytically. Also, typical graph algorithms with techniques like pointer doubling fail for two reasons. First, the graph is typically of size $n = 2^{64}$ and more, and thus cannot be constructed in memory. Second, even if it could be constructed, the graph could not be explored completely as an algorithm with at least linear complexity may take longer than our lifetime.

Parallel algorithms have been devised [4] that can explore such a graph completely, even if it cannot be constructed in memory. Yet, if the graph's size renders a complete exploration infeasible, they can also be used to "scan" such a graph by sampling. One randomly chooses $k < n$ nodes, determines the WCCs they belong to (and detects the cycles in those components), and then extrapolates this result to the complete graph. As determining the WCC of one node may already need time $O(\sqrt{n})$, this restricts the sample size and thus the validity of the extrapolation. Our contribution is an algorithm that reduces this overhead to $O(1)$, and hence allows a much larger sample of nodes to be visited.

The remainder of this article is organized as follows. In Section 2 we briefly review the relevant facts and the previous work. In Section 3 we present the new algorithm. Section 4 reports on preliminary performance results and on findings for some generators. Section 5 concludes.

2 Relevant Facts and Previous Work

State transition graphs as defined in the introduction are called *mappings* in the literature. An example graph for $n = 16$ can be seen in Figure 1. It consists of two WCCs of sizes 12 and 4 with cycle lengths of 5 and 3. For functions f randomly chosen from the set of all functions from N onto itself, Flajolet and Odlyzko [5] have derived expected values for the size of the largest component (about $0.76n$), of the largest tree (about $0.5n$), the expected path length from any node to a cycle (about $\sqrt{n}$), the expected cycle length and the expected length of the longest cycle (both $O(\sqrt{n})$, with slightly different constants close to 1).

If one starts at a node x (called *starting node*), the only thing than can be done to find out which WCC it belongs to, is to follow the unique path starting in x, by repeatedly computing $x := f(x)$, until a cycle is reached. As there is only one cycle per WCC, and there is a unique node with smallest number on the cycle (called *cycle leader*), the number of that node uniquely characterizes the component. One can find out to be on a cycle by storing after a number of steps which node has just been reached (*marker node*), and checking after each step, whether any of the stored marker nodes has been reached again. If the distances between marker nodes are always doubled, $O(\log n)$ marker nodes suffice and the effort is only increased by a constant factor [3]. As the average path length

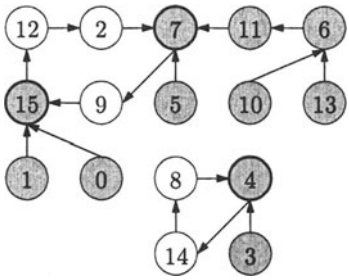

Fig. 1. An example graph.

and cycle length are both $O(\sqrt{n})$, a complete exploration has expected time $O(n\sqrt{n})$. This algorithm can be parallelized trivially, but even then the runtime is prohibitive. One can improve the runtime by keeping a store, as large as the main memory of all processors, for nodes called *pebbles* from which one already knows to which WCC they belong. Then if one reaches a pebble on a path, one can stop there.

We define a subset of the nodes called the *candidate* set. Only candidates can become pebbles. The candidate set is normally chosen independently of the function f, and in a manner that the membership to the candidate set can be computed efficiently from the node number, e.g. by requiring that some bits must be zero.

Several relationships between candidates and pebbles are possible. If every candidate indeed is a pebble, and if all pebble information is gathered in advance, then we have a completely static situation. Then in every step, we only have to check whether a candidate is reached, which can be done efficiently, and if so, we already know that we have reached a pebble. However, as we have to expect that a fraction of $1/e$ of the nodes are leaves [5], and as the candidate set is defined independently of the function f, we have to expect that $1/e$ of the pebbles are leaves as well. Pebbles on leaves are not worthwhile because only a single path can reach that pebble: the path originating in the leaf itself. If the candidate set is the set of all nodes, and if the pebbles are set during the exploration of the graph, then we have a completely dynamic situation. In this case, only a small fraction of the candidates can indeed become pebbles, because otherwise we would need $\Omega(n)$ memory resources to store the pebble information. While this scenario allows to place pebbles in a manner that takes into account the particular characteristics of the function f, it has a certain disadvantage. When following a path from some starting node, one has to check whether a pebble is reached after each step. Checking whether a candidate is a pebble requires a query to a data structure such as a search tree or a hash table and thus takes some time.

Therefore, we decided on a compromise [4]. Not every node can become a pebble. The size of the candidate set is only a fraction of n, typically $1/2^c$ as we

check whether the c lowermost bits of a node number are zero. The pebbles are placed while the graph is explored. To take the function f into account, we place pebbles in regular distances along the paths that we follow. This ensures that pebbles are spread out over the trees. The pebble data structure can be updated regularly to remove pebbles without visits in a certain time frame (similar to the LRU strategy in caches), which also gives room to place further pebbles. The pebble data structure can even be distributed over all processors to allow a pebble set that scales with machine size when we use a parallel cluster computer to follow many paths simultaneously [6].

However, the current approach suffers from a weakness. If n is really large, we cannot afford to explore the graph completely, because the effort to do so is $O(n \cdot l)$ if l is the average path length to a pebble[1]. What one can do is to restrict to a sample of $k < n$ starting nodes, chosen randomly among all nodes. If k_i of these k starting nodes belong to WCC i, then with standard techniques one can compute a confidence interval $[n_i - \delta_i : n_i + \delta]$ around $n_i = n \cdot k_i/k$ such that the size of WCC i lies in this interval with probability p. The effort for this sampling is still $O(k \cdot l')$, where l' is the average path length for the first k starting nodes. Normally, $l' > l$, as the pebbles could not yet be placed as well as after certain update improvements.

3 Efficient Sampling

We want to improve the algorithm of the previous section by taking into account the following observation: while for each starting node of the sample, only this node is attributed to a WCC, one has visited many nodes of this WCC! The bad thing is, that it is not clear how many of those nodes we have already visited in previous runs. If a path from a starting node reaches a pebble, we do not know how many of those nodes are on a path that has been visited in the past. Hence we extend our pebble data structure in order to be able to find this out.

Each pebble a now contains links all pebbles, from which paths reach a. Those pebbles are called *child pebbles*. Furthermore, we require that the tree root must be a pebble (so that we can guarantee that each path in the tree reaches a pebble), and also that each previous starting node is a pebble[2], if it contributed any newly visited starting nodes. In the unlikely event that a new starting node lies on a path visited before, it will not contribute any newly visited node and hence need not be considered further. If those requirements are maintained with every following starting node, then we can formulate the following invariant: *The pebbles, and the nodes on the paths between them, contain exactly the set of nodes already visited.*

[1] In the overwhelming majority of cases, a path ends in a pebble. Only in a tiny fraction of cases, a cycle is reached.

[2] This requires that only nodes that are candidates are chosen as starting nodes. This is however not a serious restriction as the set of candidates will be much larger than the set of starting nodes.

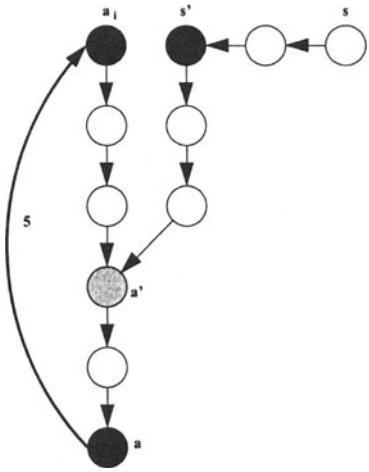

Fig. 2. Reaching a pebble.

Now, if the path from a new starting node s reaches a pebble a, we have only to find out where it met a path already visited. To do this, we visit the child pebbles of a, named a_1, a_2,..., which are in distances $d_1, d_2, \ldots$ from a. We will assume that the distances are in decreasing order. Now we find a node s' on the path from s to a in distance d_1 to a. If the distance from s to a is d (we assume $d > d_1$), then we can follow the path from s for $d - d_1$ steps. However, as we store marker nodes on the way, the effort normally is much smaller. Now we follow the paths from s' and a_1 step by step until they both reach the same node a'_1, where the paths meet. We do the same for all other child pebbles. If d' is the maximum distance of any a'_i from a, then the path from s to a contributes $d - d'$ newly visited nodes. If s is now made a pebble, and a further child pebble to a with distance d, then the invariant is maintained.

Figure 2 depicts an example situation. When starting from node s, the pebble a is reached after 7 steps. Pebble a has a pebble child a_i in distance 5. To find a node s' on the path from s to a with distance 5 to a, one starts at node s and follows the path for $2 = 7 - 5$ steps, to reach node s'. Now the paths from a_i to a and s' to a are followed simultaneously step by step, until both paths reach the same node a' after 3 steps. Hence, on the path from s to a, $5 = 2 + 3$ nodes have been visited for the first time. The node s will be made another child pebble of a with distance 7.

The overhead, defined as the number of evaluations of function f for computing d' is $O(d)$, if the number of child pebbles to a pebble is not more than a constant. This however can be achieved by adapting the pebble data structure (introducing new pebbles at places where paths from child pebbles meet) without changing the invariant. If we assume that the path length d from the

starting point to the pebble is not more than a constant factor longer than the number $d - d'$ of the newly visited nodes, the overhead will also be $O(d - d')$, and hence the overhead per newly visited node will be $O(1)$. The latter assumption is based on the fact that only a small fraction of the nodes will be pebbles and already visited nodes, and hence from a randomly chosen node, one will have on average have a long way to go until a path between pebbles is met. An additional overhead has to be accounted for the case where a starting node lies on a path that is already visited, and where no newly visited nodes will be contributed. However, if one assumes that the size n of the state space is so large that less than $10^{-3}n$ nodes can be visited in total, then the probability for this event is smaller than 10^{-3} and hence this event is seldom.

One may also argue that if a WCC i has a size n_i, then a starting node was chosen from this WCC with probability n_i/n, and thus the nodes attributed to WCC i were done so independently in the original algorithm. If the new algorithm chooses a starting node s from WCC i, then it attributes $d - d'$ nodes to that WCC, the newly visited nodes on the path starting in s. Yet, the average path length will depend on the placement of the pebbles, which will not directly depend on the WCCs, and so will the path length. Furthermore, if WCC i has had more visited nodes (in relation to its size) than other WCCs, then one has to take into account that the probability to choose an unvisited node from WCC i will sink below n_i/n, and the other WCCs will receive accordingly more starting nodes and thus more visited nodes, so that the balance is approached again.

4 Experimental Results

We have programmed a simple, sequential version of the new algorithm. As only values of n up to 10^7 are used for evaluation purposes, we could use a variant where a bit could be stored for each node, to find out whether this node was visited before. Thus, the algorithm immediately knows how many nodes have newly been visited on this path. The pebbles are placed randomly for the sake of simplicity, which will lead to a constant average distance between pebbles on a path. As overhead, we only counted the way from the first visited node to the next pebble. If we assume that on average, a path from a starting node will meet a known path in the middle between two pebbles, and that a pebble on average has two child pebbles, then we would have to increase the overhead by a factor of 6, because the length would double, and three paths would have been followed (two starting in child pebbles, one starting on the new path).

4.1 Performance Results

We tested our algorithm on a number of functions generated randomly with the help of the `lrand48` pseudo random number generator, with different seeds. We first tested functions of size $n = 10^6$. Figure 3 depicts the average number of nodes newly visited, and the corresponding average overhead, for a sequence of starting nodes. The x-axis represents the starting nodes as percentage of n, i.e.

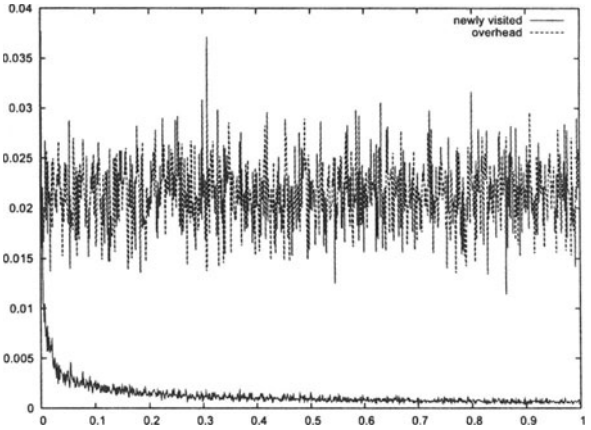

Fig. 3. Newly visited nodes and overhead per starting node for $n = 10^6$, in percentages of n.

1% of all nodes has been used as starting nodes, and the y-axis represents newly visited and overhead nodes also as percentage of n. Figure 4 presents the integral of the functions from Figure 3. We clearly see that after a certain threshold, only few nodes per path are added, while the overhead increases very much. Hence, our improved algorithm should only be applied up to this threshold. As the functions from Figure 4 can be computed while the algorithm is executed, the algorithm can stop automatically when a certain threshold is reached.

Figure 5 presents a detailed view of Figure 4 for $x \leq 0.05\%$. One sees that in this region, which is a more realistic application scenario than to use 1% of all nodes as starting nodes, the algorithm performs much more favorably.

To find out how the algorithm scales with increasing n, we plot the newly visited nodes and overhead for $n = 10^7$ in Figure 6. We see that this function looks as before, and conclude that the algorithm scales well. The same holds for the sum of visited nodes and summed overhead, which is omitted due to space restriction.

4.2 Generator Properties

For reference, we first investigated two functions that are known for a long time, one unbroken and one broken. We started with the Data Encryption Standard (DES) (see e.g. [7]), which has been a standard blockcipher from the seventies till today, although it has been replaced officially by the AES (Advanced encryption standard). The DES is a Feistel cipher with 16 rounds. In each round, one half of the 64-bit codeword is combined with a 48-bit round key by a round function. The round function contains a non-linear part. First, by doubling some of the bits of the code word (so-called expansion permutation), it is increased from 32

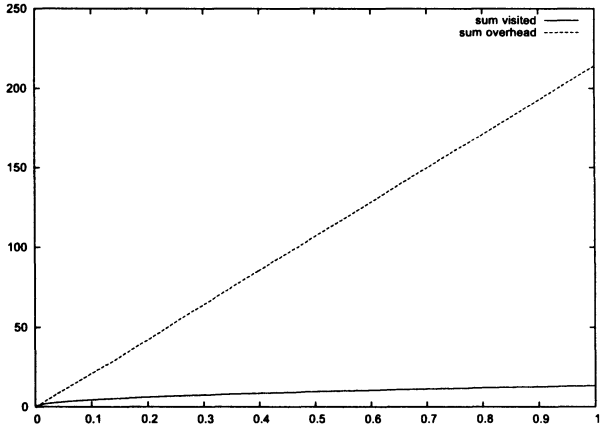

Fig. 4. Sum of visited nodes so far and summed overhead for $n = 10^6$, in percentages of n.

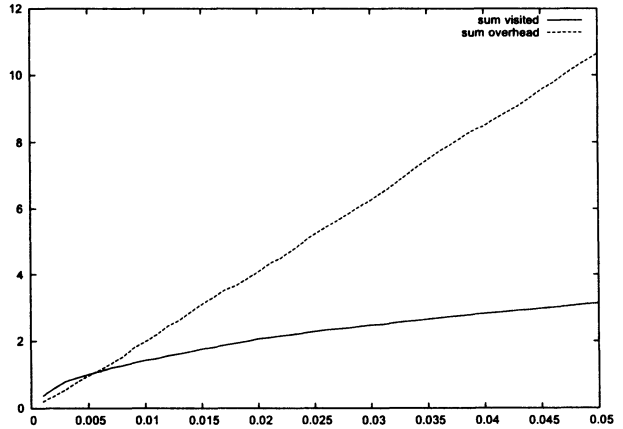

Fig. 5. Detailed view of Fig. 4 for $x \leq 0.05\%$.

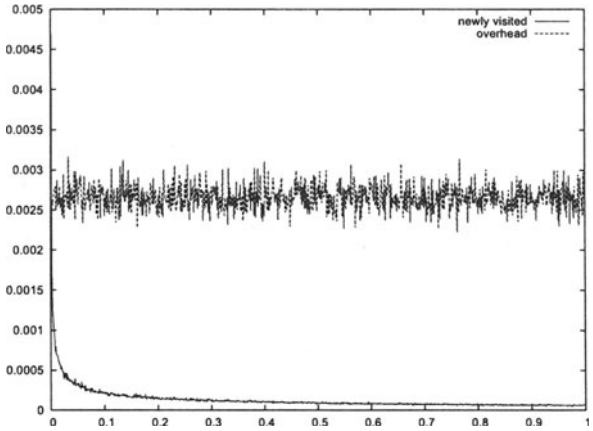

Fig. 6. Newly visited nodes and overhead per starting node for $n = 10^7$, in percentages of n.

to 48 bits. This is followed by the S-box transformation back to 32 bits. There are eight different non-linear S-boxes, each with a 6-bit input and a 4-bit output. As the expansion permutation followed by the S-box transformation is the only non-linear part of DES, and the one protecting DES against differential crypt analysis, we chose this part as a kind of generator transition function. Because of the small size $n = 2^{32}$, this graph could be explored completely. It revealed 11 WCCs (16 would be expected), the largest WCC had a size of $0.8n$ ($0.76n$ would be expected), and an average cycle length of $0.73\sqrt{n}$ ($0.63\sqrt{n}$ would be expected). Hence, this graph looks quite as expected.

Second, we took a pseudo random number generator based on a cellular automaton by Stephen Wolfram [8], which is already known to be predictable [9]. The cellular automaton consists of k linearly connected cells, each being either in state 0 or 1. Each cell's next state is dependent on its own state and the state of its neighbours. Thus, the automaton can assume $n = 2^k$ states. For an automaton with $k = 24$ cells, i.e. $n = 2^{24}$, we revealed 49 WCCs, the largest WCC having a size of $0.94n$, with an average cycle length of $42.5\sqrt{n}$. Also this graph could be explored completely. Compared to the expected values, there are too many WCCs, the largest WCC is much too large, and the longest cycle is much longer than expected.

Finally, we explored the A5/1 generator. It consists of three coupled linear feedback shift registers (LFSR) of lengths 23,22, and 19. Each LFSR has a clock bit. In each cycle, there is a majority vote over the three clock bits, and the registers with clock bits corresponding to the majority are clocked, i.e. each register is clocked in 3 out of 4 cases. The history of A5/1 is quite strange. It has been designed by ETSI (European Telecommunication Standards Institute), but

not been laid open for public scrutiny. We follow the presentation in [1] which refer to other sources that re-engineered the algorithm in a GSM mobile phone and finally got confirmation from GSM about the algorithm. Wagner et. al. also present an attack on this stream cipher, hence the security is not clear, and we felt it to be a good test case.

The state of the generator consists of the concatenated contents of the three LFSRs, thus $n = 2^{64}$. With 326,131 starting nodes we detected 59,661 WCCs. At most 103 starting nodes belonged to one WCC. The largest cycle found had a length of $0.13\sqrt{n}$. A 32-CPU cluster needed one week to compute this result. Hence, the graph looks definitely non-random. Most cycles have length $(4/3) \cdot (2^{23} - 1)$, i.e. they are defined by the period of the longest LFSR, and its frequency of clocking! Similar observations, albeit not with respect to the number of WCCs are made on slide 8 of [10].

5 Conclusions and Future Work

We have presented an algorithm that allows to explore large random graphs better than previous methods. We applied this algorithm to reveal the graph structure of several generators in the cryptographic field. Our next aim is to refine and tune our algorithm, and to explore a larger part of the state graph of A5/1, because the preliminary results indicate a quite unusual structure. Our feeling is that the surprising structure of the A5/1 state graph may also give rise to a further distinguishing attack on the A5/1 output stream.

References

1. Biryukov, A., Shamir, A., Wagner, D.: Real time cryptanalysis of A5/1 on a PC. In: Fast Software Encryption Workshop 2000, Springer LNCS (2001) 1–18
2. Eberspächer, J., Vögel, H.J., Bettstetter, C.: GSM — Global System for Mobile Communication. 3rd edn. Teubner-Verlag (2001)
3. Keller, J.: Parallel exploration of the structure of random functions. In: 6th Workshop on Parallel Systems and Algorithms (PASA), VDE (2002) 233–236
4. Heichler, J., Keller, J., Sibeyn, J.F.: Parallel storage allocation for intermediate results during exploration of random mappings. In: 20th Workshop Parallel Algorithms, Strctures and System Software (PARS). GI (2005) 126–134
5. Flajolet, P., Odlyzko, A.M.: Random mapping statistics. In: EUROCRYPT '89, Springer LNCS (1990) 329–354
6. Heichler, J., Keller, J.: A distributed query structure to explore random mappings in parallel. In: 14th Euromicro Conference on Parallel, Distributed and Network-based Processing. IEEE CS (2006) 173–177
7. Schneier, B.: Applied Cryptography. Wiley (1995)
8. Wolfram, S.: Cryptography with cellular automata. In: Proc. Crypto '85, Springer LNCS (1985) 429–432
9. Meier, W., Staffelbach, O.: Analysis of pseudo random number sequences generated by cellular automata. In: Proc. Eurocrypt '91, Springer LNCS (1991) 186–189
10. Gong, G.: ECE 710 Sequence design and cryptography (Fall 2005) lecture slides. http://calliope.uwaterloo.ca/~ggong/ECE710T4/lec8-ch6b.pdf

Mandatory Access Control applications to web hosting

Marco Prandini, Eugenio Faldella, Roberto Laschi

DEIS, University of Bologna
Viale Risorgimento 2, 40136 Bologna, Italy
{mprandini, efaldella, rlaschi}@deis.unibo.it

Abstract. "Hosting" represents a commonplace solution for the low-cost implementation of web sites through the efficient sharing of the resources of a single server. The arising security problems, however, are not always easily dealt with under the Discretionary Access Control model implemented by traditional operating systems. More robust separation between the hosted sites, as well as more robust protection of the host system, can be attained by exploiting the features typical of Mandatory Access Control systems. Recently, these systems have recently been made available to the vast Linux community through projects like SELinux and grsecurity. This paper describes the architecture of a secure hosting server, integrating SELinux functionalities into the Apache/PHP platform, designed with the goal of increasing security without adding administrative burdens or impacting performance.

1. Introduction

The vast majority of web sites do not justify the costs associated with the installation and administration of a dedicated server, permanently and reliably connected to the Internet. Since the performances of modern hardware allow to easily support many sites on a single server, *hosting* has become a widespread solution.

A web hosting server is usually based on a simple multi-user OS, allowing different *webmasters* to securely access a reserved share of the server's disk, where site contents are placed. Setting the proper permissions is not a trivial task, because the web server software runs as a particular user, which needs read access to every share.

Malicious users can exploit the ability of the server process to access resources other than their own ones by uploading, if allowed, *dynamic* pages, i.e. programs which are executed by the web server upon request of a specific URL.

The solution to this access control problem can be pursued by improving the corresponding security mechanisms of the operating systems. This work is focused on one of the most successful platforms for dynamic web sites, i.e. a Linux box running PHP [1] pages through the Apache web server [2], and describes a framework for securing the hosting server by leveraging the features provided by the Security-Enhanced Linux project (SELinux) [3]. In the following sections, the hosting scenario is examined in greater detail, the SELinux architecture and relevant features are briefly explained, and the proposed solution is discussed.

2. Hosting: security and performance issues

Server resource utilization can obviously be optimized by sharing as many components as possible among the different sites. Different solutions can be adopted, ranging from the installation of virtual machines within the host to the configuration of a single application to act as many virtual servers. Usually, there is a trade-off between security and efficiency, because isolation requires duplication of some functions.

One significant advantage of a single process serving many virtual sites is, in fact, the ability to efficiently pool available resources. Memory pages, file and network descriptors, logging facilities and so on can be optimally handled and dynamically allocated among threads or subprocesses. This solution requires either giving the web server process, running as a particular but unprivileged user, the permission to read the files of each site, or running the web server process as a privileged user, able to switch identity to the specific account of the site it is serving. Both solutions exhibit evident potential security weaknesses.

At the other end of the spectrum, virtual machines offer an excellent level of isolation, but duplicate the entire operating system. They are more convenient than a dedicated server for an organization in need of a complete set of network services (e.g.: web, e-mail, DNS, collaboration tools, ...), yet the associated costs are still not viable for the vast majority of small sites. Consequently, this approach will not be discussed further, and the following sections will illustrate single-process based solutions.

2.1. Virtual hosting of dynamic sites with Apache

The Apache web server powers the vast majority of sites worldwide [4]. It exhibits a modular structure [5, 6] allowing to extend the core functionalities so as to support any need in terms of multiprocessing models, access control, diagnostics, HTTP handling, content negotiation, virtual hosting, and execution of server-side programs. It supports two different models of server-side execution: Common Gateway Interface (CGI) and server-side scripting.

2.1.1. CGI performance and security issues

CGI [7] is the oldest model for the implementation of dynamic pages on web servers. Basically, as its name suggests, it defines a standard interface between the web server process and an external program it invokes. The web server acts almost as a pure gateway, passing all the data received from the browser to the external program, and passing the program's output back. No intervention is needed on the web server in order to attain isolation between virtual sites, because all the potentially harmful operations are carried out by the external programs.

The Apache module *mod_suexec* invokes these programs by means of a set-user-id wrapper program that executes them with site-specific credentials. In this way, another potentially exploitable component is introduced, that is the wrapper executing with root privileges, but there is no identity change of the caller web server process, a procedure that would have a negative impact on performances, as better explained at the end of the next section.

2.1.2. Server-side scripting performance and security issues

The vast majority of dynamic pages are mostly composed of fixed HTML elements, with some dynamically-generated information inserted in between. CGI programming can be uselessly cumbersome in this scenario: the program has to deal with HTML templates, generation of basic HTTP headers and other details, which often prevail over its specific function. For this reason, *server-side scripting* has been developed as a newer model for the implementation of dynamic pages. It works by tightly integrating the language used to program dynamic behaviour with the web server, making the latter able to parse a page while serving it. When, at some point in a page, the server recognizes the special tags marking embedded instructions, it substitutes them with the results of their execution, at the same point within the data flow.

This approach has known a great success, since it makes the development of dynamic web sites much easier, especially for simpler ones. ASP (Active Server Pages) on Microsoft Internet Information Server and PHP (PHP Hypertext Processor), especially popular on the Linux/Apache platform, are two examples of this technology.

However, from the security point of view, isolation between sites is more difficult when this approach is chosen over CGI, because the scripting engine becomes actually part of the web server. The only possibility would be allowing to assign different user ids to child processes spawned to serve requests belonging to different virtual servers. This is a complex modification to the server code (there is a project for an Apache module named *mod_perchild* but it is not usable) and it could still lead to an inefficient resource allocation in large installations encompassing hundreds or thousands of sites. In fact, once a child process has been given a specific id, it cannot change it anymore, so it can be reused for the same site only. If it is idle, and resources are needed to serve requests related to a different site, the only possible action is to destroy it and spawn another child from the main process.

2.2. PHP and suPHP

According to the aforementioned observations, a solution providing both the security advantages of the CGI model and the practicality of server-side scripting at the same time appears to be desirable.

The *suPHP* project [8] pursues this goal, and consequently it has been chosen as the building block of the system proposed in this paper. It exploits a peculiar property of PHP (probably the most commonly used language for the implementation of dynamic web sites [9]), that is the possibility of deploying it within Apache either as a module implementing the server-side scripting functionality, or as a CGI program.

Similarly to suexec, suPHP is composed of two parts: a small Apache module which exposes the same functionalities of the full PHP module to the web server, and a wrapper program which invokes the actual PHP interpreter in CGI mode. These components take care of invoking the PHP interpreter so that (1) PHP pages written according to the server-side scripting model are correctly processed without changes, and (2) its credentials can be changed accordingly to the involved virtual site.

While suPHP represents a valuable starting point, it should be noted that any approach based on a simple identity change of the process in charge of generating a dynamic page leaves many potential security problems open. A strict minimum-

privilege policy cannot be enforced under the discretionary access control model implemented on Linux, and consequently many options are available to a malicious webmaster for trying to compromise the server's security or simply misuse the server resources. However, there are freely available systems allowing to effectively enforce very sophisticated security policies, which can prevent most, if not all, of the possible exploits a rogue process can attempt. SELinux has been chosen in this work, being probably the most theoretically sound and widespread one.

3. Security-enhanced suPHP

3.1. SELinux basic concepts

SELinux is an implementation [10] in the Linux operating system of the Flask [11] architecture, which encompasses two components. The *Security Server* contains the definition of every security policy, and takes decisions accordingly. The *Object Managers* (one for each OS subsystem) enforce the policies by querying the Security Server for each relevant action. Four main different models cooperate to define the access control mechanism implemented by the Security Server [12]: Type Enforcement (TE) [13], Role Based Access Control (RBAC), User Identity (UI), and Multi Level Security (MLS).

The TE subsystem is the most relevant to the proposed application. According to the TE model, each subject on the system has an associated security attribute called *domain*, and similarly each object on the system has a *type*. Interactions between subjects, or actions performed by subjects on objects are controlled by an access control matrix stating the rights of a given domain when dealing respectively with another domain or a type. It is necessary to note that *classes* are used to group objects with similar access methods. The *access vector*, listing the possible kinds of access rights for an object, is not directly associated with the object but with its class, so each access control mechanism (like the aforementioned matrix) is defined in terms of which actions can be performed on a specific pair (object-type, object-class) rather than on the object-type only.

Within SELinux, actually, each subject/object is associated with a label called *security context* composed of three attributes (user, role and type), whose meaning is clarified in Table 1. Yet, the Security Server bases its *security decision* (i.e. decides whether granting the subject access to the object or not) on the third field of the context only (*type* being a synonym of *domain* for subjects), working under a closed-world policy, meaning that permissions not explicitly granted are denied.

Table 1. - Meaning of the security context attributes

	For a process	For a file
User	SELinux user who started the process	SELinux user owning the file
Role	Role associated with the user at process start time	Not used (always set to *object_r*)
Type	Domain which the process runs within	Type associated to the file

The configuration of SELinux, according to the described models, encompasses two actions: the labelling of each object with the right security context, and the definition of type enforcement policies.

The labelling step is quite simple; a file with the *.fc* (file contexts) standard extension lists, on each line, a triple composed of:

1. a regular expression matching an absolute filename,
2. a type (regular file, directory, block special, character special, socket),
3. a security context.

The definition of policies, written in a file with the *.te* (type enforcement) standard extension, is much more complex. Several heterogeneous kinds of statements can appear in a policy, like: definitions of types and rules for changing types of subjects and objects at runtime, access rules implementing the access control matrix, role declarations, role transition rules, user definition and user-to-role mapping

3.2. Design of base policies for domain operation

The most common usage of SELinux within many Linux distributions is to confine each service, such as Apache, to a specific domain. However, as already noted, granting site-specific child processes the same capabilities of the main server is not desirable, since dynamic page generation usually requires much lower privileges. Enforcing the minimum privilege principle is particularly important when wrapper programs which momentarily gain root identity are involved.

To achieve both proper privilege reduction and isolation between virtual sites, the implemented prototype defines a different domain for each virtual site, and calls for a domain transition from the Apache starting domain to the site-specific one.

In order to highlight the relevant capabilities needed by the web server, and to make policies independent from the specific distribution, a specific *apache_suphp_t* domain has been defined for standard Apache operation. Since security decisions depend on the domain only, no new users and roles have been defined, using *system_u* and *system_r* which are the default for executing system daemons. Simple file context and policy declarations, not shown for the sake of brevity, allow the *apache_suphp_t* domain to access all the relevant Apache+suPHP subsystem files, give the server the capabilities needed to properly access the essential system resources, and make every domain associated with a virtual site able to access log and configuration files.

The specific policy for the concession of proper capabilities to each domain is more complex. Moreover, every time a new virtual site is added, the corresponding policy must be added to SELinux configuration. Consequently, theoretical analysis and experimental validation was initially performed on a single site, leading to the policy definition for a specific domain, then the policy file was rewritten as a template (shown in Fig. 1), by inserting a formal *<<<Domain>>>* parameter which is substituted with the actual value by an installation script.

This template does not contain the essential rules allowing the domain transition from *apache_suphp_t* to *SuPhp_<<<Domain>>>_t*, which are detailed in the following three sections according to the three different ways to implement the transition itself (two requiring the modification of suPHP code in order to make it SELinux-aware, and one exploiting the configurable SELinux automated labelling).

```
# role and type declarations
role system_r types SuPhp_<<<Domain>>>_t;
type SuPhp_<<<Domain>>>_t;
type SuPhp_<<<Domain>>>_document_t;

# association between declared types and their function
domain_type(SuPhp_<<<Domain>>>_t)
files_type(SuPhp_<<<Domain>>>_document_t)

# access to system libraries and standard capabilities
[ omitted, similar to the main Apache ones ]

# allow a domain to read its own configuration file
allow SuPhp_<<<Domain>>>_t SuPhp_conf_etc_t:file { getattr read };

# allow a domain to access pages for its own site
allow SuPhp_<<<Domain>>>_t SuPhp_<<<Domain>>>_document_t:dir { search getattr read };
allow SuPhp_<<<Domain>>>_t SuPhp_<<<Domain>>>_document_t:file { getattr ioctl read };

# allow logging
allow SuPhp_<<<Domain>>>_t SuPhp_log_t:file { create append };
allow SuPhp_<<<Domain>>>_t var_log_t:dir { search write add_name };
type_transition SuPhp_<<<Domain>>>_t var_log_t:file SuPhp_log_t;

# allow IPC with the main Apache process
allow SuPhp_<<<Domain>>>_t apache_suphp_t:dir { getattr search };
allow SuPhp_<<<Domain>>>_t apache_suphp_t:fd use;
allow SuPhp_<<<Domain>>>_t apache_suphp_t:fifo_file { read write getattr };
allow SuPhp_<<<Domain>>>_t apache_suphp_t:process sigchld;
allow apache_suphp_t SuPhp_<<<Domain>>>_document_t:dir { getattr search };
allow apache_suphp_t SuPhp_<<<Domain>>>_document_t:file { getattr read };
allow apache_suphp_t SuPhp_<<<Domain>>>_t:process { sigkill signal };

# allow the execution of the php interpreter
allow SuPhp_<<<Domain>>>_t bin_t:file { execute execute_no_trans getattr read };
allow SuPhp_<<<Domain>>>_t bin_t:dir { search getattr };
```

Fig. 1. Template for the virtual site policies

3.3. Module-invoked domain transition

The suPHP module, when loaded, becomes part of the running Apache process, and handles the creation of the child suPHP wrapper process when needed. A simple modification to the module code allows to invoke the SELinux API function *setexeccon()* just before child creation (Fig. 2), setting the context the wrapper process is created within to the domain associated to the virtual site to be served. Reuse of the Apache process for other requests is guaranteed by resetting the context to the original value after the external execution has ended. This is the optimal strategy, because the wrapper never runs in any other domain than the site-specific one. It requires modifying a code portion which runs within the Apache process, so theoretically, if its implementation is flawed, it could add a vulnerability to the whole server. However, the patch is literally composed of 20 lines of code (error checking included), thus a thorough security revision should not be overly difficult. In order to make it work, the

Apache process must be explicitly allowed to call the *setexeccon* function and the transition from the *apache_suphp_t* domain to the *SuPhp_<<<Domain>>>_t* domain must be allowed for the suPHP wrapper executable, as shown in Fig. 3.

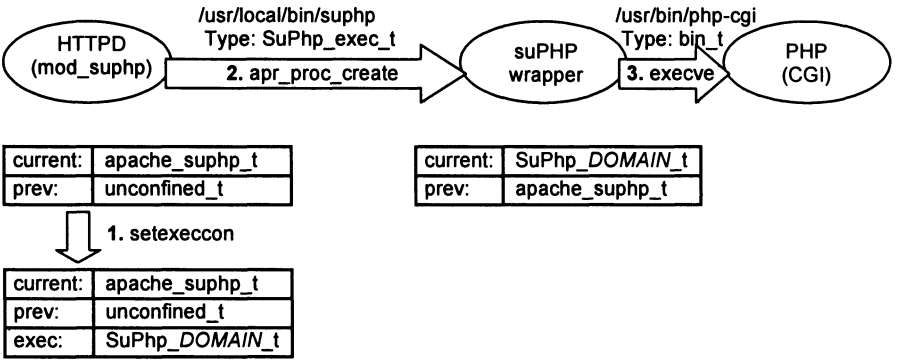

Fig. 2. Execution context evolution for module-invoked domain transition

```
allow apache_suphp_t self:process setexec;
domain_trans(apache_suphp_t, SuPhp_exec_t,    SuPhp_<<<Domain>>>_t)
```

Fig. 3. Policies for module-invoked domain transition

3.4. Wrapper-invoked domain transition

An alternative transition model encompasses the invocation of the *setcon()* function, available within the SELinux API as well. When called from within the suPHP wrapper, it causes the context transition of the process, which leaves the *apache_suphp_t* domain to enter the SuPhp_<<<*Domain*>>>_*t* domain (Fig. 4), much like the *setuid()* and *setgid()* system calls cause the process to assume the identity related to the virtual site to be served.

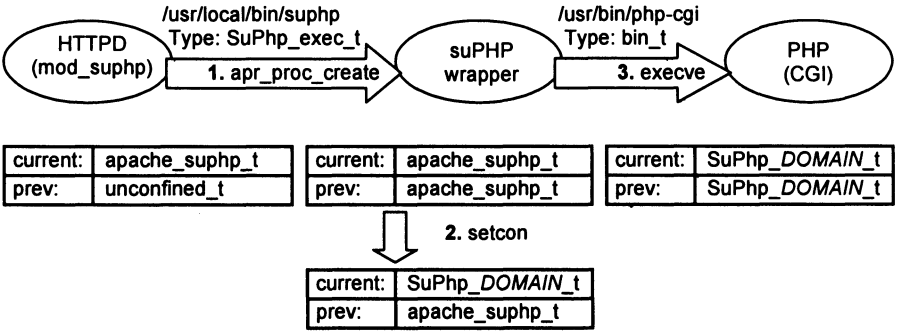

Fig. 4. Execution context evolution for wrapper-invoked domain transition

This approach has the only, negligible advantage of modifying a component which is not part of the main Apache process. However, it requires passing the final domain (read from the Apache configuration file) to the wrapper process through the environment, and needs a slightly more complex policy setup allowing the dynamic context change for the suPHP wrapper process (Fig. 5).

```
allow apache_suphp_t self:process setcurrent;
allow apache_suphp_t SuPhp_exec_t:file execute_no_trans;
allow apache_suphp_t SuPhp_<<<Domain>>>_t:process    dyntransition;
```

Fig. 5. Policies for wrapper-invoked domain transition

3.5. Policy-driven domain transition

The *domain_auto_trans* macro, when used in a policy file, instructs SELinux to execute a program within a specified domain, instead of inheriting the domain of the caller. The domain is determined by the policy file according to the type associated to the executable file (Fig. 6). This feature can be exploited to attain the correct domain transition for the wrapper without introducing changes to either the module or the wrapper itself. There is a drawback: since a file can only be labelled with a single type, in order to attain transitions to different domains (one for each virtual site) it is necessary to create a separate copy of the wrapper executable for each one.

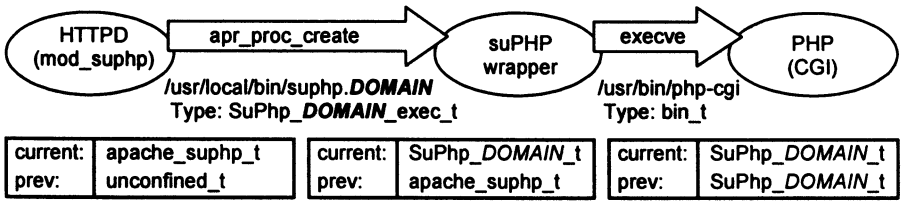

Fig. 6. Execution context evolution for policy-driven domain transition

4. Implementation

4.1. Configuration notes

The first and second approaches described in the preceding section need the domain associated with the virtual site to be passed as a parameter to the suPHP module. As any Apache module, suPHP declares its configuration directives, so that, when the module is loaded, Apache can recognize them in the configuration file, perform a formal check of their syntax, and make the associated values available to the module.

The original suPHP module declares a suPHP_UserGroup directive, specifying the standard Unix identity associated with the virtual site. The code has been modified in

order to declare a suPHP_SeLinux directive too. The added directive allows to specify only the SELinux domain which the suPHP wrapper will run within. As anticipated, all the instances will use the same user and role, because these attributes are not considered by the Security Server when making security decisions. A proliferation of useless users and roles would only make policy definition exponentially more complex.

The third approach needs the filename pointing to the correctly labelled wrapper copy to be passed as a parameter to the suPHP module. A simple convention regarding filenames allows to avoid unnecessary duplications: the wrapper copy which, by virtue of its type and of SELinux configuration, is going to run within a given *domain* has to be named /path/to/wrapper/suphp.*domain*. In this way, the module can compute the filename from the already introduced parameter suPHP_SeLinux.

4.2 Performance evaluation

The performance of the secured web server has been tested for each of the implementation schemes, and compared to the performance of a clean, non-secured server. A simple response time test was performed, by requesting a page which executes only the phpinfo() function using the Apache Benchmark (ab) tool. Fig. 7 shows the times needed for receiving the 80% of the awaited responses, for different values of concurrency, i.e. the number of simultaneously issued requests. As attended, the intervention of the complex SELinux components is visible, yet acceptable: the increase in the response time is usually below 5% for the most interesting of the three implementation variants (module-invoked domain transition).

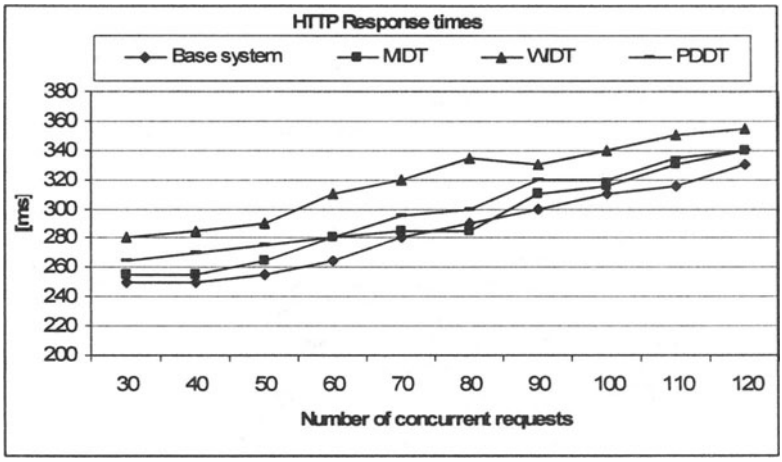

Fig. 7. Results of tests performed on non-secured system, a system implementing module-invoked domain transition (MIDT), a system implementing wrapper-invoked domain transition (WIDT) and a system implementing policy-driven domain transition (PDDT)

5. Conclusions

The presented work shows a practical application of the powerful access control model implemented by SELinux. The chosen applicative context, in the author's experience, presents significant security issues that are often overlooked for the sake of easy (lazy?) administration, but nonetheless very difficult to solve with the traditional operating systems approaches.

In fact, one of the most difficult configuration activities within these systems consists in choosing a layout of ownerships and permissions allowing each webmaster to work with its files, making them readable by the web server, but keeping them confidential with respect to other sites. Any small mistake can either prevent a site from working or make its sensitive data easily available. The problem is exacerbated by the DAC system, allowing each user to change the permissions of its own files, either maliciously or mistakenly. The adoption of MAC policies can lead to great benefits, both for achieving accurate protection of the host system and for isolating the guest virtual sites from each other.

The described system has been fully implemented and tested, both verifying the correspondence between attended and real behaviour of the domain transition policies, and measuring the impact on performance, obtaining satisfying results on both fronts. Current work is aimed to extend the analysis to the alternative systems (for example: grsecurity [14]), verifying the feasibility of defining a common framework for the configuration of secure multi-user web servers on different platforms.

References

1. PHP website - http://www.php.net/
2. Apache Server website - http://httpd.apache.org/
3. National Security Agency. Security-Enhanced Linux - http://www.nsa.gov/selinux/
4. Netcraft Web Server Survey - http://news.netcraft.com/archives/web_server_survey.html
5. Apache: Conceptual Architecture by Ahmed Hassan - http://plg.uwaterloo.ca/~aeehassa/cs746/as1/apache1.htm
6. Extending Apache: Apache Modules - http://f-m-c.org/projects/apache/html/3_3Extending_Apache.html
7. Common Gateway Interface v1.1 - http://hoohoo.ncsa.uiuc.edu/cgi/
8. suPHP Project by Sebastian Marsching - http://www.suphp.org/
9. PHP usage stats - http://www.php.net/usage.php
10. Implementing SELinux as a Linux Security Module, by Stephen Smalley, Chris Vance, Wayne Salamon - http://www.nsa.gov/selinux/papers/module.pdf
11. R. Spencer, S. D. Smalley, P. Loscocco, M. Hibler, D. Andersen and J. Lepreau, The Flask Security Architecture: System support for diverse security policies, *Proc. 8th USENIX Security Symposium*, Washington, D.C., 1999, pp 123-139
12. S. D. Smalley, Configuring the SELinux Policy. *Nai Labs Report #02-007*, June 2002
13. L. Badger, D. F. Sterne, D. L. Sherman, K. M. Walker and S. A. Haghighat, A Domain and Type Enforcement Unix Prototype, *Proc. 5th USENIX UNIX Security Symposium*, Salt Lake City, UT, 1995, pp 127-140
14. grsecurity website - http://www.grsecurity.net/

Outsourcing Security Services for Low Performance Portable Devices

Attila Szentgyörgyi[1], András Korn[1]

[1]Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, Magyar Tudósok körútja 2, 1117 Budapest, Hungary
{szentgyorgyi.attila,korn.andras}@tmit.bme.hu

Abstract. The number of portable devices using wireless network technologies is on the rise. Some of these devices are incapable of, or at a disadvantage at using secure Internet services, because secure communication often requires comparatively high computing capacity. In this paper, we propose a solution which can be used to offer secure network services for low performance portable devices without severely degrading data transmission rates. We also show that using our approach these devices can utilize some secure network services which were so far unavailable to them due to a lack of software support. In order to back up our claims, we present performance measurement results obtained in a test network.

Keywords: wireless, portable device, security, security proxy, security services

1 Introduction

Nowadays, an increasing number of people use Personal Digital Assistants (PDAs) and smartphones. Alas, the limited processing power of some of these devices leads to severe performance degradation when using strong encryption, required for secure on-line communications. Also, some network security services (e.g. some VPN technologies) are completely unavailable to the users of these devices because the service provides no client support for their platform.

While these problems might disappear in the future as portable devices become more powerful, it's also possible that the trend for ever smaller devices persists, so that there may always be a class of devices for which the computational overhead of strong encryption poses a challenge.

The performance problem we attempt to address is grave. While a PDA we tested was able to achieve network throughput of 0.163 MB/s using no encryption, this dropped to 0.036 MB/s when we switched to IPSec [7,8]: a drop of about 78%. The higher CPU load can also lead to drastically reduced battery life.

We propose to outsource the computationally expensive parts of network traffic encryption to a security proxy (SP), a trusted device provided by the network operator.

Users who wish to take advantage of this service may instruct the security proxy to establish a secure communication channel with a specified address and forward

unencrypted traffic from the user's device to the other encryption endpoint through the encrypted tunnel (and vice versa, of course). We show that it is possible to do this without the user having to surrender authentication secrets.

In the rest of the paper, we introduce the security proxy architecture in detail, including the communication protocol used between the proxy and the portable device and the proxy and other network devices.

Measurement results that demonstrate the effectiveness of the security proxy approach are also included.

2 Architecture of the security proxy

As noted above, the security proxy takes care of establishing an encrypted connection to a remote server and then forwards packets between the portable device and the remote server. Since traffic between the security proxy and the portable device is unencrypted, the portable device must trust the network and the security proxy in order to use this service; however, if the other option is trusting the entire Internet, the choice should be easy.

The security proxy has two distinct modes of operation: *tunnel* mode and *gateway* mode.

In *tunnel* mode, an encrypted tunnel is created and arbitrary IP packets can be routed through it. This is ideal for VPN applications. The security proxy acts as a VPN client and establishes a VPN connection with the remote endpoint specified by the portable device (authentication needs to be taken care of; see below).

In order for the portable device (PD) to be able to send packets through the VPN, all routers between the security proxy (SP) and the PD must be made aware that all or some packets from the PD must be routed towards the SP. See below for details.

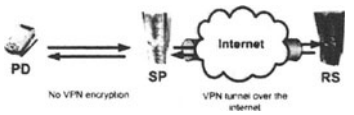

Fig. 1. Security proxy tunnel mode, for VPN applications

In *gateway* mode, the security proxy acts as an application layer gateway, translating a weak protocol like telnet or pop3 into one with strong encryption, such as ssh or pop3s.

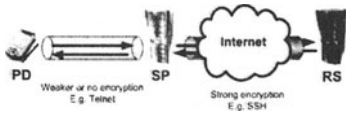

Fig. 2. SP gateway mode for gatewaying e.g. telnet to SSH.

In *gateway* mode, the PD initiates an application-level connection with the SP using some weak protocol like telnet; the SP in turn establishes a corresponding

encrypted connection to the real server (in the case of telnet, using SSH). The SP internally translates all telnet messages into SSH messages and SSH replies into telnet replies, so that the user of the PD can effectively telnet into an SSH server.

This usage also requires routing changes. If the routers don't know about the SP, they will forward telnet packets to the ssh server directly; and if the user telnets to the SP instead, the SP won't know what server to connect to using SSH, because it will only see its own address in the destination field of the TCP SYN packet the PD sends. Therefore, the routers must again be told to route some or all packets from the PD through the SP. This way the SP can transparently proxy the telnet connections and turn them into SSH connections.

The fundamental difference between the two modes of operation is which network layer they work in. In tunnel mode, the PD can send arbitrary IP packets through the encrypted connection. In gateway mode, a specific application layer protocol is gatewayed into another one, and no data packets can be sent through the encrypted connection directly. This, alas, also means that gateway support for every application layer protocol must be implemented separately.

3 Outsourced authentication

The proxy must be able to authenticate itself on behalf of the user to remote servers. A straightforward but insecure way to do this would be to require the PD to surrender its authentication secret to the SP. The obvious disadvantage of this approach is that the SP could then later establish connections on behalf of the user at will.

Therefore, we propose *outsourced authentication*, where the SP forwards authentication challenges from the contacted server to the PD and replies from the PD to the server; thus, the user needn't surrender his or her secret, and the only computationally expensive operation the PD has to carry out is replying to the authentication challenge. This procedure can be repeated as many times as necessary.

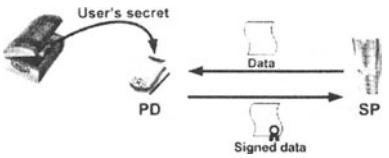

Fig. 3. Outsourced authentication. The SP sends the authentication information to be signed, mobile device signs it with the user's secret and sends back the signed data to the SP.

4 Routing issues

As mentioned earlier, the security proxy must receive the to-be-encrypted traffic from the portable device with the destination IP address intact (that is, with the actual destination IP, not the IP of the SP, in the header).

One way to accomplish this is to place the security proxy in the normal traffic path between the PD and the rest of the network; that is, either pointing the default route through it or placing it directly behind the access device the PD is connected to.

This has several drawbacks:

- The SP needs higher routing performance than it would need if it only saw traffic it actually needs to process in some way.
- In a network with many access devices, establishing this topology may be impractical.
- It is more difficult to set up load balancing between several security proxies.

It is also possible to selectively route packets to the SP by setting up appropriate routing policies in the routers between the PD and the SP. While this alleviates the problems mentioned above, it introduces additional complexity by requiring the SP and the routers to communicate with each other.

Operators willing to implement a security proxy can decide whether the additional complexity is worth the increased scalability and flexibility. The Security Proxy Control Protocol (SPCP) we propose includes the primitives necessary to modify routing policies, but their use is not mandatory.

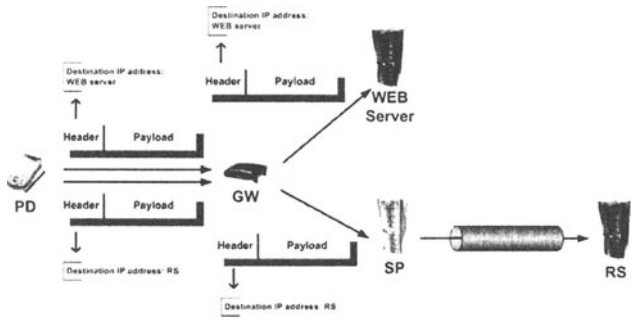

Fig. 4. Gateway (router) controlled by SP. The routing decision is based on the source and destination address of the packets, not just the destination.

In *tunnel* mode, the PD can specify a list of IP ranges that should be routed through the encrypted tunnel. The SP then instructs the router(s) between itself and the PD to route packets that originate from the PD and are addressed to any of these ranges through the SP.

This doesn't introduce a security risk because the PD can only influence the routing of its own packets, not arbitrary packets. The additional processing load on the routers may have to be considered though.

In *gateway* mode, each to-be-proxied service has a well defined IP address, protocol and port number. The router(s) must be instructed to route matching packets towards the SP. For example, if the SSH server at 1.2.3.4 is to be reached using the telnet protocol, TCP packets sent by the PD and bound for 1.2.3.4, port 23 – or, if the telnet service itself must stay reachable, port 22 – should be routed towards the SP. All other packets to 1.2.3.4 use the default route.

Some routers may not be able to base their routing decision on protocols and port numbers; in this case, several workarounds are possible, e.g.:

- An IP-over-IP VPN (using e.g. GRE) may be set up between the PD and the SP. Because there is no encryption, no substantial processing overhead should result. Note that this approach completely eliminates the need to modfy routing between the PD and the SP in any way and is thus also a viable option if the routing management elements of the SP concept are not deployed.
- All packets addressed to the selected remote servers could be routed to the SP, which could then decide which ones it needs to proxy and which can be forwarded as-is.

5 Security Proxy Control Protocol

In order to facilitate communication between the SP and the PD (and optionally between the SP and the router), we specified the SPCP application layer protocol with extensibility in mind. Syntactically and conceptually, SPCP resembles HTTP and SIP. It can take care of connection establishment, outsourced authentication, routing change requests and connection termination.

5.1 Communication between the portable device and the security proxy

SPCP uses TCP as its transport protocol. A client session lasts as long as the TCP connection between the PD and the SP is alive (keep-alives must be sent periodically). The SP discards all state information and undoes routing changes related to the client if the TCP connection is terminated. This mechanism ensures that no "dangling" routes are left over and that a new client that assumes the IP of a previous one cannot use the services requested by the previous client.

The protocol uses simple request-response semantics. Each message consists of a number of attribute=value pairs. The response to a request-type message is always a status message.

The client initiates the connection by sending a CONNECT message. This message could later be extended to include attributes useful for authentication; client profiles could be stored on the proxy and automatically activated on successful login.

The client can request a new encryption service or deactivate an old one by sending a MODIFY message with a subtype of *set* or *del* respectively (which will modify the in-memory service profile associated with the client, hence the name).

The client receives a status message from the proxy only after all necessary initialization has been taken care of; for example, in the case of gateway mode, the SP first sets up the appropriate routing rules before indicating success (or failure) to the client. This keeps the protocol simple because there will only ever be one status reply to any request.

If an authentication process needs outsourced authentication, the SP sends a "201 authentication required" STATUS message that contains the authentication challenge in an *authparam* attribute. The PD then sends back the original MODIFY message extended with the response to the authentication challenge.

An UPLOAD message is provided to transmit relatively large chunks of data, such as cryptographic certificates, to the proxy [1]. These can then be re-used during authentication to remote services and need not be re-uploaded again. Note that thanks to outsource authentication, it is not necessary to upload the private key.

A well-behaved client should send a DISCONNECT message before disconnecting; this ensures that the proxy can de-allocate all reserved resources, discard state information and reset routing changes related to the client without having to wait for the TCP keepalive to time out. The advantage for the client is that it can be certain that the next device to obtain its IP address will not be able to use its encrypted tunnels even for a short time. Note that this window of insecurity can be eliminated completely if the client is required to use a GRE tunnel to communicate with the SP.

5.2 Communication between the security proxy and the router

The ROUTE protocol primitive can be used to instruct the router to add or remove routing rules to selectively route packets towards the SP.

Naturally, the router must be able to authenticate the SP in some way, otherwise allowing an external device to affect routing would be a security risk.

As above, the router and the SP maintain a TCP connection as a means of ensuring soft-stateness; if the connection is broken, all routing rules requested by the corresponding SP should be discarded.

5.3 Example: establishing an IPSec session

Let's take a look at how a PD might set up an IPSec session with a remote VPN concentrator through the SP. Figure 5, below, shows the required message sequence. All communication (except between the SP and the remote server RS) uses SPCP.

Notice how the security proxy acts both as an SPCP server for the PD as well as an SPCP client to the router (GW).

The first message is the CONNECT message of the PD. The SP might preload a saved service profile associated with the user after successful authentication of the user; this is beyond our scope at this point.

The SP acknowledges the CONNECT message by sending an "OK" status message.

Next, the PD sends a MODIFY set request, specifying the type of service requested (IPSec), the address of the remote server, and optionally several IP/netmask pairs to be routed through the tunnel.

At this point, the SP begins setting up the IPSec session with the remote server. It completes the first four messages of the IKE [2] phase, but in order to construct the fifth message the IPSec client running on the SP needs the user's private key. The SP replies to the MODIFY request of the client with a status of "201 authentication required", and includes the HASH_I value the client must sign with its own secret key.

The client signs HASH_I using the RSASSA-PKCS1-v1_5-SIGN [3] primitive and repeats the MODIFY set request, including the signed HASH_I value needed to complete authentication.

The SP inserts the signed HASH_I value into the SIG_I payload and sends out IKE phase one message five. The remote server accepts the signature because it was created with the user's private key.

The IPSec server replies with a signed HASH_R value packed in the SIG_R payload. The SP can check the validity of SIG_R because the IPSec server certificate is public and includes the public key. If the authentication process is successful, the second phase of IKE can commence.

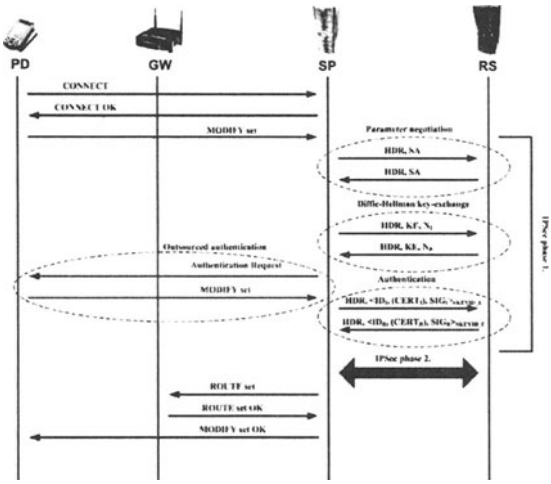

Fig. 5. IPSec service setup sequence diagram.

When the IPSec session between the IPSec client running on the SP and the remote server is established, the SP sends a ROUTE message to the default router of the PD to set up routing as requested by the client. If the router is able and willing to comply, it replies with a STATUS of ROUTE set OK.

Finally, the security proxy replies to the last MODIFY set message of the client with a STATUS message of MODIFY set OK, indicating that the service is ready to use.

If the mobile device wants to request a new service in addition to this one it simply sends a new MODIFY message with the parameters of the new service.

6 Security issues

If the PD and the SP want to authenticate each other, the entire conversation can be secured using e.g. SSL and certificate-based authentication. Alternatively, authentication can take place in the CONNECT and CONNECT OK message so the

security proxy and the mobile device can authenticate each other at the beginning of the connection. Message authentication codes can be placed in all other messages too.

However, on an untrusted network, any number of attacks would be possible against the communication between the PD and the SP (in addition to exposing private, not-yet-encrypted data as it travels between these two entities). Therefore, network or data link layer security must be present at all points between the PD and the SP.

One way to accomplish this is to use hardware link layer encryption between the PD and the router (e.g. WPA-TKIP or WPA-AES CCMP [6]) and some strong VPN such as IPSec between the router and the SP. We note that although WPA-PSK-TKIP is vulnerable to brute force attacks, strong passwords still provide adequate security.

Fig. 6. In a secure network, packets must only travel through encrypted and authenticated channels.

Scalability is important; after all, the whole idea of a security proxy is meant to enhance the performance of portable devices. Therefore, it is always advisable to use encryption that the devices in question have hardware support for.

Because sensitive data is going to traverse it in the clear, the PD must be able to trust the network. When using WPA in "Enterprise mode", the PD can authenticate the access point and the entire network in a certificate-based manner. This prevents the "evil twin" man-in-the-middle attack where the attacker impersonates an access point and is thus recommended for running a security proxy.

IP spoofing must be prevented as well; otherwise if client A sets up an encrypted tunnel, client B might be able to send packets into that tunnel by spoofing the address of client A (or even receive client A's packets using ARP poisoning). Luckily, in a WPA-TKIP environment, spoof protection is almost automatic. A portable device cannot spoof its MAC address because that would break the encryption. Thus, spoofing both the MAC and the IP address, which would be difficult to detect, is impossible.

Preventing IP spoofing can be done at the access point. If – as is the case in most networks – address assignment is done using DHCP, the AP can monitor DHCP traffic and set up static ARP entries based on DHCP address assignments. Packets whose source IP-MAC combination doesn't match the ARP table can be discarded. This also defeats ARP poisoning. We have a prototype of such a filter running under the OpenWrt[9] Linux distribution on an off-the-shelf access point.

7 Results

We developed a prototype of the proxy and performed measurements to demonstrate its performance. We used a WPA Enterprise network in which a RADIUS [5] server authenticated the user using EAP-TLS [4]. We used an HP IPAQ H5550 PDA

(Personal Digital Assistant) with a Java based client application. SP client, IPSec client and OpenVPN client ran on a Pentium 4 computer.

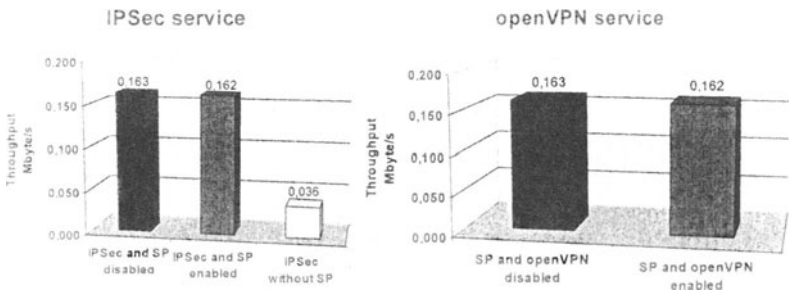

Fig. 7. Test results

In the tests the portable device utilized an IPSec service with and without the help of a SP. Using the SP, transmission speed was almost as high as with no encryption, as expected. Without the SP, the IPAQ was able to achieve a mere 22% of the plaintext throughput. Additionally, using the proxy, we were able to communicate with an OpenVPN server even though no officially supported OpenVPN client is available for Windows Mobile at this time. Other VPN solutions that don't support PDA clients can be made accessible to PDAs as well.

8 Summary

We have shown that it is possible and feasible to enable low-performance portable devices to use strong network encryption by introducing the security proxy that takes care of the computationally expensive operations on behalf of the portable device.

We presented security proxy architecture in details, including the SPCP used between the proxy and the portable device and the proxy and other network devices. Additionally we introduced the outsourced authentication method with which security proxy can authenticate itself on behalf of the user without the knowledge of user's secret. Finally the realization of security proxy proved that data transmission speed was the same as in the no encryption scenario which was four times faster than PDA IPSec encryption.

The technology can also be used to gateway between weak and strong protocols, so that even services with no client support on portable devices become accessible.

9 Acknowledgements

The authors would like to thank Ákos Csiszér for his work on the DHCP-based IP spoofing filter.

References

[1] R. Housley, W. Ford, W. Polk, D. Solo: *Internet X.509 Public Key Infrastructure, Certificate and CRL Profile*, RFC 2459, 1999
[2] D. Maughan , M. Schertler, M. Schneider, J. Turner: *Internet Security Association and Key Management Protocol (ISAKMP)*, RFC 2408, 1998
[3] B. Kaliski, J. Staddon: *PKCS #1: RSA Cryptography Specifications*, RFC 2437, 1998
[4] B. Aboba, D. Simon: *PPP EAP TLS Authentication Protocol*, RFC 2716, 1999
[5] Jonathan Hassel: *RADIUS*, O'Reilly, 2002
[6] Jon Edney, William A. Arbaugh: *Real 802.11 Security: Wi-Fi Protected Access and 802.11i*, Addison-Wesley, 2003
[7] R. Thayer at al.: *IP Security Document Roadmap*, RFC 2411, 1998
[8] S. Kent, R. Atkinson: *Security Architecture for the Internet Protocol*, RFC2401, 1998
[9] OpenWrt project homepage, online, http://www.openwrt.org/

Public Verifiable Multi-sender Identity Based Threshold Signcryption

Wen Chen, Feiyu Lei, Fang Guo, and Guang Chen

Department of Communications and Electronic Engineering,
School of Information Science and Technology,
Donghua University,
1882 Yanan Road, Shanghai, 200051
wen.chenwen@gmail.com

Abstract. In this paper, we present a new identity based signcryption scheme with public verifiability using quadratic residue and pairings over elliptic curves, and give a security proof about the original scheme in the random oracle model. Furthermore, this paper focuses on a multi-sender(t,n) identity based threshold sighcryption. Finally, we prove the scheme of threshold setting is secure as the original scheme.

Keywords: identity based, threshold, signcryption, public verifiability

1 Introduction

In this paper, we present a Multi-sender (t, n) identity based threshold signcryption with public verifiability, which contains three important primitives: threshold cryptography, signcryption, identity based systems.

1.1 Threshold Cryptography

Threshold cryptography [11] provides for increased security of the distributed platform by distributing protocols among a number of participants. A t out of n threshold signature scheme is a protocol that an adversary who corrupts at most $t-1$ players (the adversary knows all the secrets of corrupted players) cannot obtain any available information about the secret key of the system or forge a valid signature. Each player should compute his partial signature. After we collect at least t valid partial signatures, we can combine these partial signatures to an original signature.

1.2 Identity Based Signcryption with Public Verifiability

In 1997, Zheng [1] proposed a new primitive called signcryption, which is more efficient than the conventional 'sign-then-encrypt' approach. A secure signcryption scheme should satisfy confidentiality, unforgeability, and public verifiability (including non-repudiation).

Definition 1: *When a signer Alice denies her signature, the receipt Bob can prove that Alice is just the signer of her signature in an efficient and secure way. If Bob doesn't compromise his secret key and the plaintext to anyone, we say that the signcryption scheme satisfy* ***Public Verifiability*** .

In addition, we note that public verifiability should not affect the confidentiality and unforgeability of the scheme.

1.3 Related works and Our Contributions

The combination of threshold scheme and identity based signcryption leads to identity based threshold signcryption scheme. This problem has not yet been well solved in the literature, although some threshold signcyption schemes [2, 3, 12] have been proposed. They only consider the unsigncryption process in a (t, n) threshold manner. The previous schemes can be regarded as multi-recipient signcryptions, that can be unsigncryptable by multiple designated verifiers.

The focus of our study is on a multi-sender threshold signcryption with public verifiability. And the main contributions are as follows:

Firstly, based on LQ's scheme [15], we propose an efficient signcryption with public verifiability. As a result we adopt this signcryption as the original signcryption of our threshold scheme.

Next, we give a multi-sender (t, n) threshold signcyption model, which has multiple senders and only a recipient. After collecting the valid at least t partial signcryptions, the recipient can compute the original signcryption. Our threshold work builds on the above original signcryption and secret sharing [5]. In this paper, we combine and simplify two protocols to achieve secure threshold signcryption. Compared with traditional threshold signature-then-encryption, the scheme achieves more attractive features, such as high efficiency and low communication overheads.

Lastly, the scheme can easily find the corrupted party, which is an excellent property in threshold cryptography.

2 Preliminaries

2.1 Pairings and Quadratic Residue

Pairings: Let $(\mathbb{G}_1, +)$ be a cyclic additive group generated by P, whose order is a large prime p, and $(\mathbb{G}_2, \cdot)$ be a cyclic multiplicative group with the same order p. Let $e: \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ be a map with the following properties:

1. **Bilinearity:** $e(a \cdot P,\ b \cdot Q) = e(P, Q)^{ab}$ for all $P, Q \in \mathbb{G}_1, a, b \in \mathbb{Z}_p$;
2. **Non-degeneracy:** There exists $P, Q \in \mathbb{G}_1$ such that $e(P, Q) \neq 1$;
3. **Computability:** There is an efficient algorithm to compute $e(P, Q)$ for $P, Q \in \mathbb{G}_1$.

We use the supersingular elliptic curves and Weil Paring [16] to realize such groups. The security of the scheme depends on the hardness of Bilinear Diffie-Hellman problem and Decisional Bilinear Diffie-Hellman problem [16].

Definition 2. *Given two groups* $\mathbb{G}_1$ *and* $\mathbb{G}_2$ *of the same order* q*, a bilinear map* $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ *and a generator* P *of* $\mathbb{G}_1$*, the* ***Bilinear Diffie-Hellman (BDH)*** *problem is to compute* $e(P,P)^{abc}$ *given* (P, aP, bP, cP)*. the* ***Decisional Bilinear Diffie-Hellman (DBDH)*** *problem is, given a tuple of points* (P, aP, bP, cP) *and an element* $h \in \mathbb{G}_2$ *to decide whether* $h = e(P,P)^{abc}$ *or not.*

The Decisional Bilinear Diffie-Hellman problem is not harder than Bilinear Diffie-Hellman problem. No algorithm is known to be able to solve any of them so far.

Quadratic Residue: [17]Let N be an integer, and $Z_N^* = \{k \in Z_N | (k, N) = 1\}$. $a \in Z_N^*$ is said to be quadratic residue modulo N, if there exists an $x \in Z_N^*$ such that $x^2 \equiv a \ mod \ N$. It is infeasible to compute from $x^2 \equiv a \ mod \ N$ if modulo N can't be factorized.

2.2 Protocol Emulation

The common idea to prove threshold security [4] [6] [8] is that the adversary $\mathcal{A}$'s view in the threshold setting can be simulated by a simulator θ that runs in the original scheme.

3 Multi-sender Threshold Signcryption Model and Security Requirements

3.1 System Model

Communication Model. We consider a set of n senders $\{P_1, ..., P_n\}$, indexed $1, ..., n$, and a recipient *Bob*, and a static adversary $\mathcal{A}$ who can corrupt the set of n senders.

The Static Adversary. If the set of corrupted parties is fixed before the protocol begins, we call the adversary non-adaptive or static. Assumed that the static adversary can corrupt up to $t-1$ of the n players, such as $n \geq 2t+1$.

System Parameters. Initially, given security parameters k , the Trusted Key Generator (TKG) chooses a groups G_1 of prime order q , a generator P of q, and chooses F_q^* as G_2, a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$, three hash functions $H_1 : \{0,1\}^* \rightarrow G_1$ and $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^n$ and $H_3 : \{0,1\}^n \times \{0,1\}^n \rightarrow F_q^*$. Then, TKG chooses a master key $c \in F_q^*$ and computes $P_{pub} = cP$. Besides, TKG chooses a large RSA modulus N that nobody can efficiently factorize. The algorithm pair (E, D) is secure symmetric encryption and decryption algorithms respectively. The system's public parameters are: $K = (G_1, G_2, n, e, P, P_{pub}, H_1, H_2, H_3, N)$, where n denotes the size of ciphertext. Given an identity ID, TKG computes the public key $Q_{ID} = H_1(ID) \in G_1$, then sets $d_{ID} = cQ_{ID}$ to be the private key . Alice's key pair is (Q_{ID_A}, d_{ID_A}). Bob's key pair is (Q_{ID_B}, d_{ID_B}). Note that we choose the same system parameters as in the Malone-Lee scheme [?] except a large secure modulus N.

3.2 Security Requirements

The original signcryption should satisfy the following properties: IND-IDSC-CCA, EF-IDSC-ACMA, and Public Verifiability [15] [18] [1]. We have given the definition of Public Verifiability in Section 1. Next the definitions of IND-IDSC-CCA, EF-IDSC-ACMA are given in **Definition 3.** and **Definition 4.** respectively.

Definition 3. *The scheme is secure against adaptive chosen-signcryptext attacks that is defined in the game **(IND-IDSC-CCA)** played between a challenger $\mathcal{B}$ and an adversary $\mathcal{A}$, if no polynomially bounded adversary has a non-negligible advantage ϵ.*

Definition 4.2. *The scheme is said to be secure against an existential forgery for adaptive chosem messages attacks **(EF-IDSC-ACMA)** if no polynomially bounded adversary has a non-negligible advantage in the following game.*

Then, we consider that the threshold signcryption should satisfied the following properties [9] [4] [6] [8]:

Simulatablity. As described in Section 2.2, if the threshold signcryption is simulatable, the threshold signcryption is secure as the original signcryption.

Robustness.The robustness of (t, n) threshold signature scheme means that an static adversary who corrupts less than $t-1$ players should not be able to prevent uncorrupted players from generating valid signatures.

4 The Proposed Original Signcryption

Based on LQ scheme [15], we present an identity based signcryption with public verifiability. Our scheme is more efficient than LQ scheme. We show the signcryption and unsigncryption as follows.

Signcryption of m by Alice the Sender:	sends (C, R, S) to Bob
chooses $x \in_R [1, ..., q-1]$	Unsigncryption of (C, R, S) by Bob the Recipient:
$v = e(Q_{ID_B}, P_{pub})^x$	$v = e(S, Q_{ID_B}) \times e(Q_{ID_A}, d_{ID_B})^R$
$w = v^4 \ mod \ N$	$t = H_2(v)$
$t = H_2(v)$	decrypts $m = D_t(C)$
$C = E_t(m)$	$w = v^4 \ mod \ N$
$R = H_3(C, w)$	Accepts m only if $R = H_3(C, w)$
$S = xP_{pub} - Rd_{ID_A}$	

If Alice denies her signcryption, Bob computes $K_1 = e(Q_{ID_B}, P_{pub})^{2x} \ mod \ N$, then forwards (C, R, K_1) to a TTP. TTP computes $w = K_1^2 \ mod \ N$, then verifies $R = H_3(C, w)$. If the equation holds, TTP says that Alice tells a lie.

5 The Proposed Threshold Signcryption

We have a set of n senders $\{P_1, ..., P_n\}$, $t-1$ is the number of corrupted players, such as $n \geq 2t+1$.

Distributed Key Generation and Joint Pedersen Verifiable Secret Sharing in [5] are easily extended to that in the identity based setting. We make use of Distributed Key Generation to generate key pairs d_{ID_A}, Q_{ID_A} and d_{ID_B}, Q_{ID_B} in the identity based setting. Joint Pedersen Verifiable Secret Sharing in the Identity Based setting is used to distribute the randomness x. Finally, each sender P_i obtains the information $(d_{ID_{Ai}}, Q_{ID_{Ai}}, x_i)$. Let *Interpolate* denote the standard polynomial interpolation. Then it is possible to compute $v = Interpolate(v_1, ..., v_t) = e(Q_{ID_B}, P_{pub})^x$ where $v_i = e(Q_{ID_B}, P_{pub})^{x_i}$.

We show the threshold signcryption and unsigncryption in the following.

Signcryption of m ***by*** P_i ***the Sender:***
$v_i = e(Q_{ID_B}, P_{pub})^{x_i}$
$w_i = v_i^4 \ mod \ N$
$t_i = H_2(v_i)$
$C_i = E_{t_i}(m)$
$R_i = H_3(C_i, w_i)$
$S_i = x_i P_{pub} - R_i d_{ID_{A_i}}$

Unsigncryption of (C_i, R_i, S_i) ***by Bob the Recipient:***
$v_i = e(S_i, Q_{ID_B}) \times e(Q_{ID_{A_i}}, d_{ID_B})^{R_i}$
$t_i = H_2(v_i)$
decrypts $m = D_{t_i}(C_i)$
$w_i = v_i^4 \ mod \ N$
Accepts m only if $R_i = H_3(C_i, w_i)$

After collects the t ***signcryptions*** (C_i, R_i, S_i), ***Bob does as follows:***
computes t valid v_i respectively.
$v = Interpolate(v_1, ..., v_t)$.
$w = v^4 \ mod \ N$
$C = E_{H_2(v)}(m)$
$r = H_2(C, w)$
$S = Interpolate(S_1, ..., S_t) \ mod \ N$
$R = r \cdot Q_{ID_A}$
Then he gets the original signcryption (C, R, S).

6 Efficiency Analysis

We consider that the most expensive operations are paring in G_1, exponentiation in G_2 (short for Exp.), and multiplication in G_1(short for Mul.) (the square evaluation modulo N, multiplications in G_2 are omitted). $|x|$ denotes the number of bits in x. We may seen the computation cost $Exp. \simeq Mul.$.

Table 1 tells us that our scheme is more efficient than LQ's scheme, although with small information expansion. Then, without loss of generality, comparing with a threshold scheme based on traditional Schnorr signature and ElGamal encryption [12], our threshold scheme is more efficient than traditional sign-and-encryption threshold schemes.

7 Security Proofs

We should consider the security of the original signcryption and the threshold signcryption as described in Section 3, respectively.

Table 1. Comparison with previous solutions

	Communication Costs	Computational cost
LQ scheme	$\|m\| + \|H\| + \|q\|$	$5parings, 2Exp., 4Mul.$
Our original scheme	$\|m\| + \|N\| + \|q\|$	$2parings, 1Exp., 4Mul.$
A traditional threshold scheme	$\|H\| + 2n\|p\| + 2n\|q\|$	$7nExp.$
Our threshold scheme	$n\|m\| + n\|N\| + n\|q\|$	$nExp.and4nMul.$

7.1 The Security of the Original Signcryption

Confidentiality: The **Theorem 1** proves that our scheme provides confidentiality in the random oracle model, which performs analogously to the proof of [15].

Theorem 1 *In the random oracle model, if an adversary $\mathcal{A}$ has a non-negligible advantage ϵ against the IND-IDSC-CCA security of the proposed scheme TSC when running in a time t and performing q_{SC} Signcrypt queries, q_U Unsigncrypt queries and q_{H_i} queries to oracles H_i (for $i = 1, 2, 3$), then there exists an algorithm $\mathcal{B}$ that can solve the DBDH problem in the group $\mathbb{G}_1$ with a probability $\epsilon' \geq \epsilon - q_U/2^k$ in a time $t' < t + (8q_{SC}q_{H_3} + 4q_U)te$, where te denotes the time required for one pairing evaluation.*

proof. Given in appendix A.

Unforgeability: The Unforgeability against an existential forgery for adaptive chosem messages attacks ***(EF-IDSC-ACMA)*** derives from Hess's identity signature scheme [13]. It is obviously that the adversary doesn't know more information about signature than that in LQ scheme. Under the BDH assumption, the unforgeability of our scheme is as secure as that of LQ scheme.

Public Verifiability: Once Bob computes $K_1 = e(Q_{ID_B}, P_{pub})^{2x} \ mod \ N$, everyone can verify the signature (R, S) of the message m. Without factorization of N, TTP also can't extract $e(Q_{ID_B}, P_{pub})^x$ in $K_1 = e(Q_{ID_B}, P_{pub})^{2x} \ mod \ N$. So given (C, R, K_1), TTP cannot compute v so that the plaintext message m can't be computed by TTP. Thereforeit is computationally feasible for any TTP to settle a dispute between Alice and Bob without divulging Bob's private key and the message.

7.2 The Security of the Threshold Signcryption

The Simulation. We assume that the original signcryption scheme is secure. The successful secure simulation gives us a proof which outputs a probability distribution that is identical to the distribution that the real adversary sees in an real execution of our threshold signcryption. We construct a successful simulator $SIMSIGENC$ as follows.

We have a set of n senders $\{P_1, ..., P_n\}$, $t-1$ is the number of corrupted players, such as $n \geq 2t+1$. $SIMSIGENC$ doesn't know the real d_{ID_A}, x, v. We make use of $SIMDKG$ to generate key pairs d'_{ID_A}, Q_{ID_A} and d'_{ID_B}, Q_{ID_B} in the Identity Based setting, where $SIMDKG$ is a perfect simulator in [5]. Joint Pedersen Verifiable Secret Sharing in the Identity Based setting is used to distribute the randomness x'. Each simulated sender P'_i obtains the information $(d'_{ID_{A_i}}, Q_{ID_{A_i}}, x'_i)$.

Signcryption of m ***by*** P'_i ***the simulated Sender:***
$v'_i = e(Q_{ID_B}, P_{pub})^{x'_i}$
$w'_i = v'^4_i mod N$
$C'_i = E_{H_2(v'_i)}(m)$
$R'_i = H_3(C'_i, w'_i)$
$S'_i = x'_i P_{pub} - r'_i d'_{ID_{A_i}} mod N$
Unsigncryption of (C'_i, R'_i, S'_i) ***by Bob the Recipient:***
$v'_i = e(S'_i, Q_{ID_B}) \times e(Q_{ID_{A_i}}, d'_{ID_B})^{R_i}$
decrypts $m = D_{H(v'_i)}(C)$

$w'_i = v'^4_i mod N$
$R'_i = H_3(C'_i, w'_i)$
Accept m only if $R'_i = H_2(C'_i, w'_i)$
After collects the t ***signcryptions*** (C'_i, R'_i, S'_i), ***Bob computes***
$v' = Interpolate(v'_i)$.
$w' = v'^4 mod N$
$C' = E_{H_2(v')}(m)$
$R' = H_3(C, w)$
$S' = Interpolate(S'_1, ..., S_t) mod N$

Clearly, we show that the simulator above outputs the simulated $(d'_{ID_{A_i}}, Q'_{ID_{A_i}}, x'_i, v'_i, x', v', C'_i, R'_i, S'_i)$ which is identical to the real information $(d_{ID_{A_i}}, Q_{ID_{A_i}}, x_i, v_i, x, v, C_i, R_i, S_i)$ of our threshold signcryption. We conclude that the security of our threshold signcrypotion is secure as the original signcryption. We refer readers to [4, 6, 8] for details about the simulation.

Robustness. In this paper, Bob receives the (C_i, R_i, S_i) only when the verification holds: Accept m only if $R_i = H_3(C_i, w_i)$. The verification detects the corrupted party easily, so that the threshold scheme provides the robustness.

8 Conclusion

Identity based threshold cryptosystem is a useful practical tool to protect system security in the open network, and signcryption is a new primitive to achieve the efficient communication and computation. In this paper, we present a new identity based signcryption scheme with public verifiability using quadratic residue and pairings over elliptic curves. Based on this work, we propose a multi-sender (t, n) identity based threshold signcryption. Both the two schemes are secure and efficient.

References

1. Y. Zheng, Digital signcryption or how to achieve cost (signature & encryption) << cost(signature) + cost(encryption), In Advances in Cryptology - Crypto'97, LNCS 1294, pp. 165 - 179, 1997.

2. Y. Mu and V. Varadharajan. Distributed signcryption. INDOCRYPT'00, LNCS 1977, pp. 155-164. Springer-Verlag, 2000.
3. D. Kwak and S. Moon. Efficient Distributed Signcryption Scheme as Group Signcryption. In: Applied Cryptography and Network Security (ACNS'03), LNCS 2846, pp. 403-417. Springer-Verlag, 2003.
4. Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. Adaptively secure multiparty computation. In Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, pages 639-648, Philadelphia, Pennsylvania, 22-24 May 1996.
5. R.Canetti,R.Gennaro, S.jarecki,H.Krawczyk, and T.Rabin. Adaptive security for threshold cryptosystems. In Advances in Cyptology-CRYPTO 99. Springer-Verlag, 1999.
6. S.Micali and P.Rogaway, Secure computaion. In Joan Feigenbaum, editor, Advances in Cryptology - Crypto'91, pages 392-404, Berlin, 1991. Springer-Verlag. Lecture Notes in Computer Science Volume 576.
7. D.Beaver, Foundations of secure interactive computing. In Joan Feigenbaum, editor, Advances in Cryptology - Crypto'91, pages 377-391, Berlin, 1991. Springer-Verlag. Lecture Notes in Computer Science Volume 576.
8. V.Shoup. Practical threshold signatures. IBM Reseearch Report RZ3121(April 19 1999).
9. Y. Desmedt, Threshold cryptography, European Transactions on Telecommunications, 5(4), 1994.
10. Y. Desmedt and Y. Frankel, Threshold cryptosystems, Advances in Cryptology, Crypto'89, LNCS Vol. 435, 1989.
11. Hyung Koo, Hyun-Jeong Kim, Ik Rae Jeong, Dong-Hoon Lee, and Jongin Lim, Jointly unsigncryptable signcryption, WISA 2001, Vol.2, pp. 397-407, 2001.
12. F.Hess. Efficient identity based signature schemes based on parings , SAC 2002, LNCS 2595, Springer-Verlag, 2003. 310-324
13. J. Malone-Lee and W. Mao. Two birds one stone: signcryption using RSA. In Topics in Cryptology - proceedings of CT-RSA 2003, LNCS 2612, pp. 211 - 225. Springer, 2003.
14. Libert B, Quisquater J J. New Identity Based Signcryption Based on Parings. Cryptology ePrint Archive, Report 2003/023, http://eprint.iacer.org, February 2003
15. Boneh D, Franklin M. Identity Based Encryption from the Weir Paring, Advances in Cryptology-Crypto'01, LNCS 2139, Springer-Verlag, 2001. 213-219,
16. Rabin M O. Digitalized Signatures and Public Key Functions as Intractable as Factorization. MIT Laboratory for Computer Science,1979, MIT/LCS/TR-212.
17. Baek J, Ron S, Zheng Y. Formal Proofs for the Security Signcryption, PKC 2002, LNCS 2274, Springer-Verlag, 2002. 80-98

Appendix A: Proof of theorem 1

DBDH distinguisher $\mathcal{B}$ receives a random instance (P, aP, bP, cP, h) of DBDH problem. $\mathcal{B}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$'s challenger in IND-IDSC-CCA game. $\mathcal{B}$ should maintain lists L_1, L_2 and L_3, which are initially empty and are used to keep track of answers to queries asked by $\mathcal{A}$ to random oracles H_1, H_2 and H_3. Any Signcrypt and Unsigncrypt query on a pair of identities happens after $\mathcal{A}$ queried the hashing H_1 of these identities. $\mathcal{A}$ never makes an Unsigncrypt query on a ciphertext obtained from the Signcrypt Oracle.

H_1 **queries:** At the beginning of the game, $\mathcal{B}$ gives $\mathcal{A}$ the system parameters with $P_{pub} = cP$. $\mathcal{B}$ doesn't known the PKG's master key c. Then, after $\mathcal{A}$ asks a polynomially bounded number of H_1 queries on identities of his choice, $\mathcal{B}$ chooses random numbers $i, j \in \{1, ..., q_{H_1}\}$. Let the i^{th} and j^{th} H_1 query's answer are $H_1(ID_i) = aP$, $H_1(ID_i) = bP$, respectively (Note:aP, bP are random instance of DBDH problem, and $\mathcal{B}$ doesn't know the values of a, b.). For queries $H_1(ID_e)$ with $e \neq i, j$, $\mathcal{B}$ chooses $b_e \leftarrow_R F_q^*$, puts the pair (ID_e, b_e) into list L_1 and answers $H_1(ID_e) = b_e P$.

H_2 **queries:** On a $H_2(g_e)$ query where $g_e \in G_2$, $\mathcal{B}$ firstly searches a pair (g_e, R_e) in the list L_2. If the pair is found, $\mathcal{B}$ answers R_e, otherwise $\mathcal{B}$ answers $\mathcal{A}$ by a random number $R \leftarrow_R \{0,1\}^n$ such that no entry $(*, R)$ exists in L_2.

H_3 **queries:** On a $H_3(c_e, k_e)$ query where $c_e \in \{0,1\}^*, k_e \in G_2$, $\mathcal{B}$ firstly searches a tuple (c_e, k_e, r_e) in the list L_3. If the tuple is found, $\mathcal{B}$ answers R_e, otherwise $\mathcal{B}$ chooses a random number $r \leftarrow_R F_q$ such that no entry $(*, *, r)$ exists in L_3. Then $\mathcal{B}$ answers (c_e, k_e, r) to $\mathcal{A}$ and puts the tuple (c_e, k_e, r) into L_3

Key extraction queries: When $\mathcal{A}$ asks Keygen(ID_A), if $ID_A = ID_i$ or $ID_A = ID_j$, then $\mathcal{B}$ fails and stops. Otherwise then the list L_3 must contain a pair (ID_A, d), this means $\mathcal{B}$ previously answered $H_1(ID_A) = dP$ on a H_1 query with ID_A. The ID_A's private key is then $d_{ID_A} = dcP = dP_{pub}$. Then $\mathcal{B}$ answers d_{ID_A} to $\mathcal{A}$.

Signcrypt queries: $\mathcal{A}$ can perform a Signcrypt query for a plaintext M and identities ID_A and ID_B.

- If $ID_A \neq ID_i, ID_j$, $\mathcal{B}$ computes d_{ID_A} according to above **Key extraction queries:**, and then run the algorithm $Signcrypt(M, d_{ID_A}, Q_{ID_B})$.
- If $ID_A = ID_i$ or $ID_A = ID_j$ and $ID_B \neq ID_i, ID_j$, $\mathcal{B}$ simulates the algorithm $Signcrypt(M, d_{ID_A}, Q_{ID_B})$ as follows ($\mathcal{B}$ *has to simulate* $Signcrypt(M, d_{ID_A}, Q_{ID_B})$ *since he doesn't know the secret key* d_{ID_i}):
  1. $\mathcal{B}$ chooses $R \leftarrow_R F_q$ and $S \leftarrow_R G_1^*$, then computes $v = e(S, Q_{ID_B}) \times e(RQ_{ID_A}, d_{ID_B})$ where d_{ID_B} is the private key of ID_B ($\mathcal{B}$ obtains it from the key extraction algorithm because $ID_B \neq ID_i, ID_j$).
  2. $\mathcal{B}$ runs H_2 simulation to find t and computes $C = E_t(m)$, $w = v^4 \ mod \ N$. If $\mathcal{B}$ finds a tuple (C, w, R') in L_3 where $R \neq R'$, he has to repeat the process with another random pair (R, S) until finding a tuple (C, w, r) whose (C, w) do not figure in a tuple of L_3. This process can be repeated at most $2q_{H_3}$ times. Each attempt must contain four pairing computation. Once an admissible tuple (C, w, R) is found , $\mathcal{B}$ puts (C, w, R) into L_3, and then answers the ciphertext (C, R, S) to $\mathcal{A}$.
- If $ID_A, ID_B = ID_i, ID_j$, $\mathcal{B}$ simulates the algorithm $Signcrypt(M, d_{ID_A}, Q_{ID_B})$ as follows: $\mathcal{B}$ chooses $v^* \leftarrow_R G_2$ and $S^* \leftarrow_R G_1^*$, then runs H_2 simulation to find $t^* = H_2(v^*)$ and computes $C^* = E_{t^*}(m)$, $w^* = v^{*4} mod N$. $\mathcal{B}$ has to check the tuple collision and repeat finding process as above second step.

Once an admissible tuple (C^*, w^*, R^*) is found , $\mathcal{B}$ puts (C^*, w^*, R^*) into L_3, and then answers the ciphertext (C^*, R^*, S^*) to $\mathcal{A}$.

Unsigncrypt queries: $\mathcal{A}$ can perform a Unsigncrypt query for a ciphertext (C', R', S') with ID_A, ID_B.

- If $ID_A, ID_B = ID_i, ID_j$, $\mathcal{B}$ always notifies $\mathcal{A}$ that the cipertext is invalid. In fact, $\mathcal{B}$ cannot decrypt the ciphertext about ID_i, ID_j since he doesn't know the private keys d_{ID_i}, d_{ID_j}.
- If $ID_A, ID_B \neq ID_i, ID_j$, $\mathcal{B}$ first computes $v' = e(S', Q_{ID_B}) \times e(R' Q_{ID_A}, d_{ID_B})$, $w' = v^4 mod N$, ,$R' = H_3(C', w')$ and checks if the list L_3 contains the tuple (C', w', R'). If no such tuple is found, $\mathcal{B}$ rejects the ciphertext. Otherwise, he searches for a query $H_2(v')$ in the list L_2. If no such query is found , $\mathcal{B}$ takes a random pair $(v', t' = H_2(v')) \in G_2 \times \{0,1\}^n$ such that $(*, t')$ already exists in L_2 and inserts (v', t') into L_2. Lastly, $\mathcal{B}$ computes $m' = D_{t'}(C')$ and answers m' to $\mathcal{A}$. If $\mathcal{A}$ previously asked the has value $H_3(C', w')$, $\mathcal{B}$ answered R' with a probability of at most $1/2^k$. So for all queries, the probability to reject a valid ciphertext does not exceed $q_U/2^k$.

$\mathcal{A}$ chooses a pair of identities on which he wishes to be challenged after he perform a polynomially bounded number of queries. With a probability at least $1/\binom{n}{k}$, the pair of target identities will be (ID_i, ID_j). We see that if $\mathcal{A}$ asks the private key of ID_i or ID_j before he choosing the target identities, $\mathcal{B}$ fails and stops. Further more, if the target pair isn't (ID_i, ID_j), $\mathcal{B}$ fails and stops.

When $\mathcal{A}$ sends two plaintexts m_0 and m_1 to $\mathcal{B}$, $\mathcal{B}$ chooses a random bit $b \in_R \{0,1\}$ and signcrypts m_b as follows:

$\mathcal{B}$ chooses $R^* \leftarrow_R F_q, S^* \leftarrow_R G_1^*$
$v^* = e(S^*, Q_{ID_B}) \times h^{R^*}$, where h is a candidate for the DBDH problem obtains $t^* = H_2(v^*)$ from H_2 simulation.
$C_b = E_t^*(m_b)$

Now we assess the probability of $\mathcal{B}$'s success. let **DBDHBrk** be the event that $\mathcal{A}$ chooses the target pair of identities (ID_i, ID_j) on which he wishes to be challenged after he perform a polynomially bounded number of queries. As long as the simulation of the attack's environment is perfect, the probability for **DBDHBrk** to happen is the same as in a real attack. In real attack, when the simulation is perfect we have

$\mathbf{Pr}[\mathcal{A}\ success] = \mathbf{Pr}[\mathcal{A}\ success \cap \neg \mathbf{DBDHBrk}] + \mathbf{Pr}[\mathcal{A}\ success \cap \mathbf{DBDHBrk}]$
$\leq \frac{1}{2}(1 - \mathbf{Pr}[\mathbf{DBDHBrk}]) + \mathbf{Pr}[\mathbf{DBDHBrk}]$
$= \frac{1}{2} + \frac{1}{2}\mathbf{Pr}[\mathbf{DBDHBrk}]$

and then we have $\epsilon = 2\mathbf{Pr}[\mathcal{A}\ success] - 1 \leq \mathbf{Pr}[\mathbf{DBDHBrk}]$. Next, the probability that the simulation is not perfect remains to be assessed. The only case where it can happen is that a valid signcryptext is rejected in a Unsigncrypt query. The probability to reject a valid signcryptext is thus not greater than $q_u/2^k$. Hence $\epsilon' \geq \epsilon - q_u/2^k$. The bound on $\mathcal{B}$'s computation time is derived from the fact that that every Signcrypt query requires at most $8q_{H_3}$ pairing, and evaluations every Unsigncrypt query requires at most 4 pairing evaluations . ■

A Discussion on the Role of Deception in Information Operations for the Defence of Computer Networks[a]

Zafar Kazmi, Theodore Tryfonas
Information Security Research Group
Dept. of Electronics & Computer Systems Engineering
Faculty of Advanced Technology
University of Glamorgan
United Kingdom
E-mail: zkazmi@glam.ac.uk, ttryfona@glam.ac.uk

Abstract. Deception has been widely deployed in human conflicts, but its application to information security could only be witnessed since the early 1990s. Different deception techniques have been proposed as part of computer-based information operations (IO), in order to facilitate specific warfare campaigns. In the recent years, a number of researchers have investigated different deception techniques used in computer-based networks, such as honeypots and honeynets, but in the light of several emerging issues more consideration is required in exploring deception as a strategic capability for Computer Network Defence (CND). In this discussion paper we address the potentially important role that deception can play for CND, similarly to its important role within conventional warfare and IO. We also highlight issues of further research towards the integration of deception in information security practices.

Keywords: Deception, information operations, information system

1. Introduction

Over the past twenty years, the increased use of computer systems and the swift boost of the Internet were accompanied by the equal growth of computer security incidents. The numerous reports on incidents highlight this assertion in a global context (e.g. CSI/FBI, 2005; E&Y, 2005 etc.). Technologies and the threats related to these are both becoming more and more complex; information and computer systems face a wide variety of threats which can result in significant damage to an organisation's vital infrastructure. In this context, it is the awareness of the threats and vulnerabilities of a particular system that allows for the selection of the most effective security measures.

The latter includes building a strong network defence by employing physical, procedural and personnel security measures as well as deploying digital security measures such as Firewalls, Anti-Viruses and Intrusion Detection Systems (IDSs). More recently this includes the deployment of deceptive techniques (Spitzner, 2003).

---

[a] An earlier draft of this paper was discussed in the 2005 Workshop on Safeguarding National Infrastructures (SNI 2005), Glasgow.

As opposed to the conventional mentioned controls that are meant to confront (e.g. by dropping data packets – firewall) or at least alert on (e.g. by intercepting suspicious communications – IDS) an intrusion, the latter depends on providing to the attacker a false sense of the target ('honeypots'). Via means of those configurations the actions of an attacker can be observed and their capabilities can be better understood.

Network defence based on those building blocks, is put in place to deal with types of attacks, such as service interruption, interception of sensitive email or transmitted data and remote (mis)use of computer resources. The design of protective measures for computational infrastructures follows usually a conventional rational, translated through a metaphorical understanding of a physical attack and defence (Tryfonas & Kiountouzis, 2003); it is like preparing for a material conflict in a digital arena.

The concept of the honeypot as mentioned above is not new in the field of study of human conflict. Indeed, other means than that of direct confrontation have been employed throughout human history, one of those concepts being *deception* (Armistead, 2004). The concise Oxford dictionary defines deception as the misrepresentation, persuasion for what is false; mislead purposely (Sykes, 1981). Deception has been well studied in the context of conventional warfare (e.g. Seymour, 1996), but as information technology becomes more and more an essential commodity in and off the terrain, there is a need to rediscover, revisit this concept and explore its implications for modern, IT-driven operations.

In this paper we investigate the role of deception for computer network defence and the opportunities for deploying it as a countermeasure for information security. The paper is structured as follows; in the following section two, we present elements of deception within information operations as we know them to today. In section three we discuss the need to revisit the concept in order to deploy deception within computing-based operations. We finally conclude with a reference to potential further research work in the area.

2. Elements of Deception in Conventional Warfare and Information Operations

There are over seventeen different definitions of Information Operations (IO). In this context, we adopt a state-of-art definition that defines IO as the strategic planning and coordination of activities necessary to protect an organisation's information (QinetiQ, 2003). Defensive IO, unlike offensive IO, are carried out in order to protect and defend information systems by introducing, integrating and co-ordinating policies, procedures, personnel and technology (Ashenden & Jones, 2003). Figure 1 outlines the different IO categories.

Documented use of IO techniques as part of warfare goes back to the 10th century BC when King Solomon said: *"A wise man has great power, and a man of knowledge increases strength; for waging war you need guidance, and for victory many advisers."* The more information one has the better he will be able to assess a situation in taking advantage of certain variables for achieving information superiority.

Deception is an act of deceiving or misleading and can also be defined as the problematic distinction between appearance and reality (Rue, 1994). It can be considered as the creation and invocation of both offensive and defensive environments and can be employed for attacking an adversary's perception of what is actually happening. Furthermore, deception can be applied to enhance an operation, exaggerate, minimise, or distort the enemy/opponent's perception of capabilities and intentions, to mask deficiencies, and to otherwise cause a desired outcome where conventional military activities and security measures were unable to achieve the desired result (Cohen & Lambert, 2001).

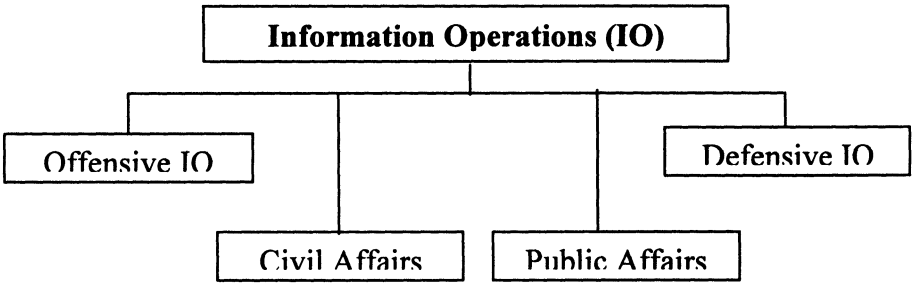

Figure 1: Information Operations Categories (JCS, 2003)

The concept of deception as a technique in conflict is ancient (Campen & Dearth, 1998). As long ago as 1469 BC, during the reign of Thutmose III, the Egyptians used different deception techniques to fool their enemies, and pass into Syria through an unsecured route (Sun, 2002). Plenty of other examples of the use of deception techniques in warfare campaigns, such as Homer's Trojan-Horse or World War II stories (e.g. misleading the German intelligence for their targeting of V-1 and V-2 missile attacks; Sun, 2002), demonstrate the important role that deception played in warfare throughout military history.

The deception used in military operations is defined in the United States Joint Doctrine for Military Deception (JCS, 1996) as:

> "*Actions executed to deliberately mislead adversary military decision makers as to friendly military capabilities, intentions, and operations, thereby causing the adversary to take specific actions that will contribute to the accomplishment of the friendly mission*"

Through deception one can manage an adversary's perception and disrupt their decision-making processes. These processes feed into the adversary's defensive INFOSEC processes which when disrupted will allow the success of offensive 'NETOPS' (Waltz, 1998). Use of deception can lead to information superiority, being the capability to collect, process, and disseminate an uninterrupted flow of information while exploiting or denying an adversary's ability to do the same (Waltz, 1998). NETWAR is the information-related conflict waged against nation states or

societies at the highest level, with the objective of disrupting, damaging, or modifying what the target population knows about itself or the world around it (Waltz, 1998).

Deployment of Psychological Operations (PSYOPS) could make IO more efficient and may also help in achieving the desired goal more rapidly. In specific, PSYOPS are planned operations to convey selected information and indicators to foreign audiences to influence their emotions, motives, objective reasoning, and ultimately the behaviour of foreign governments, organizations, groups, and individuals. The purpose of psychological operations is to induce or reinforce foreign attitudes and behaviour favourable to the originator's objectives (ibid, IWS, 2004).

Misleading information and deceptive tactics may play an important role in a successful campaign, enabling the monitoring of an opponent's moves and ensuring a desired completion of a specific military operation. The United States DoD's statement about the importance of PSYOPS in a military operation manifests that PSYOPS are a vital part of the broad range of United States diplomatic, informational, military, and economic activities (JCS, 2003).

IW Model Layer		Function	NETWAR
Offence	Perceptual	Manage perception, Disrupt decision processes	PSYOPS, Deception
	Information	Dominate information infrastructure	NETOPS
	Physical	Break things..., Incapacitate/kill people	Physical destruction
Defence	Perceptual	Protect perceptions and decision-making processes	Intelligence, Counterintelligence
	Information	Protect information infrastructure	INFOSEC
	Physical	Protect operations, protect people	OPSEC

Table 1 - Taxonomy of information operations (Waltz, 1998, p.208)

In the taxonomy of table 1, deception is an element that may have a vital role for the success of several operations, both for offensive and defensive IO. To support the observation, the United States Joint Chiefs of Staff (JCS) Memorandum of Policy (MOP) 116 Military refers to deception as a technique that has proven to be of considerable value in the attainment of national security objectives, and a fundamental consideration in the development and implementation of military strategy and tactics.

Deception allows for subduing the enemy having potentially avoided a physical conflict, thus reducing logistic costs and related resources. Furthermore, engagement in a conflict is usually a reactive action. However, in the area of computer security reacting is expensive; being proactive and preventing attacks or minimising threats is thought to be more cost-effective (Vidalis, 2004). The contemporary IO battle space may include the following elements of deception, all directed towards exploitation of high-tech means

1. National antagonism – e.g. email PSYOPS campaigns, 'netspionage' (espionage through the Internet) and Internet based open source intelligence, defence systems cracking, massive denial of service attacks etc.

2. Industrial espionage – e.g. netspionage, destruction of magnetic media, computer theft, competitor trash capture & analysis, social engineering etc.
3. Individual targeting and ID theft – e.g. social engineering, phishing, spoofing etc.

To tackle the complexity of this 'digitised' landscape, some deception techniques in computer network security emerged since the early '90s. As the purpose of deception is to surprise the adversary, in a computing environment the outcome of such an element of surprise can be twofold. Either the defenders have time to react and deploy necessary countermeasures (or activate existing ones), or the threat agent will call the attack off and return to the information gathering process in order to re-examine their plan of action. Deception can be a common operation to achieve the disruption of intelligence and counterintelligence operations of adversaries (Cohen, 1998) and thus it is believed to be the future of IO and Information Security.

An example of application of deception techniques in an IO campaign within a contemporary technology context is a United States military email campaign of urging Iraqi leaders to take over Saddam Hussein's regime (Friedman, 2003). One of the several emails stated that "*Iraqi chemical, biological and nuclear weapons violate Iraq's commitment to agreements and United Nations resolutions. Iraq has been isolated because of this behaviour*" (Friedman, 2003). The intentions of the United States and its allies were clearly for the Iraqi people to realise the injustice Saddam Hussein, the Iraqi president of that time (Friedman, 2003).

The Iraqi authorities responded to that threat by blocking the emails in order to ensure that the messages do not spread throughout the country (Friedman, 2003). As one could argue that there was no strong evidence of Iraq having chemical, biological and nuclear weapons and was perhaps based on intelligence assumptions, this example shows deployment of deceptive techniques by the US military in an attempt to win the hearts and minds of Iraqi military and civilian leaders. Hence, using this communication in order to gain advantage could be considered as misleading and therefore justifies it with the definition of deception.

3. Towards the Integration of Deception in Computer Networks

Conventional deception techniques have been studied over the years and appear in military textbooks and sources for long now. However in regard to information warfare, despite the fact that a number of different deceptive techniques have appeared as part of IO, there are not a lot of resources studying the issue in its contemporary, hi-tech context. A threat assessment carried out by the US Navy (Hildreth, 2001), indicated that nation states such as China, Russia and India are reported to have developed different policies of preparing for a cyber-warfare and are engaged in rapidly developing their IO capabilities. The report further indicates that countries such as Iran, Syria, Libya and North Korea have some IO capabilities whereas other countries such as France, Japan and Germany have comparatively advanced capabilities (Hildreth, 2001). These nation states may also be investigating defensive IO capabilities and not only exploring offensive IO capabilities.

Existing frameworks, such as Cohen & Lambert's Framework for Deception (2001), address the matter in principle, however it does not offer deployment of different deceptive techniques for a specified CNO operation. Other models, such as the cognitive model for exposition of human deception and counter-deception (Lambert, 1987), is quite generic and does not allow targeted deception in a CNO environment. This model is based on developing a basic understanding of human deception which would then lead to a comprehensive development of a framework for organising deception principles and examples. Hence a methodological construct to assist in the planning and deployment of deception as part of IO could add value to the defensive organisation of computer networks. In the following paragraphs we outline concepts that could later serve as components of a methodological construct for deploying deception with a computer network environment.

Deception can be considered as a vital element of information security and therefore can play an increasingly important role to achieve desired objectives when deployed as part of network defence. Targeted deception should:

- be coordinated with concealment of true intentions
- reinforce enemy expectations
- be integrated with operations
- have realistic timing and duration
- be imaginative and creative

The deployment of effective deception can be an important element of information and computer based system's security. More specific deception techniques have been introduced in the field of information security, to play their role in computer network defence. The deployment of Honeypots and Honeynets in a computer network can lead to the discovery of an attacker's movements and allow the network to be secured against the attacker's next offensive move and strategies. Honeypots are systems designed to be appeared as fully functioning elements of the infrastructure, placed at an appropriate location on the network where all inbound and outbound traffic is captured and monitored, providing a secure and controlled environment to allow attackers to access them (Gupta, 2003; Spitzner, 2003).

It is also possible to deploy deceptive mechanisms as a precaution and protective measure. Indeed, with the progression and maturity of computer virtualisation, virtual machines can now be used when performing operations as simple as web browsing (e.g. VMware, 2006), or as complex as the provision of a web service. With this technique, a virtual machine is used to perform the required operation, concealing all details of the actual computer. This is consequence protects the original system from threats such as exposure to malicious software, hacking etc.

Technology has allowed for an increased capability for information gathering, but perceptions and the nature of decision-making have a common vulnerability – the human factor (Mitnick & Simon, 2002). System users are probably the biggest vulnerability, as they are susceptible even to low-tech attacks, such as social engineering (Mitnick & Simon, 2002). It is people who sit behind monitors, typing and/or communicating commands; people are in charge of automated procedures and

can shut them down if they perceive that something is wrong and that the computer reactions do not make sense; and in the context of a computer network, humans are tasked with administering the systems and networks.

In respect to the latter, an indicative list of the responsibilities of network administrators may include:

- Design a network which is efficient in terms of resources
- Deploy large numbers of machines which can be easily upgraded
- Decide what services are needed
- Plan and implement adequate security
- Provide a comfortable environment for users and keep them satisfied
- Develop ways of fixing errors and problems when occur
- Keep track of, and understand how to use, new and emerging technology

Some of these tasks, by being straightforward, are vulnerable per se. For example, merely by designing a logical network and storing plans and other information about it, an administrator exposes vital information about the computing infrastructure, and an adversary can follow the same logic and enumerate it. However, an attacker suffer from the same weakness as much as the defender does; *attacking operations are meaningful to people* and hence deception may have a role here, as much as it has a role in practices such as social engineering. In this example, deception can be used to hide the real computers amongst false 'dummies'.

Figure 2 outlines a possible planning process for preparing and deploying deception.

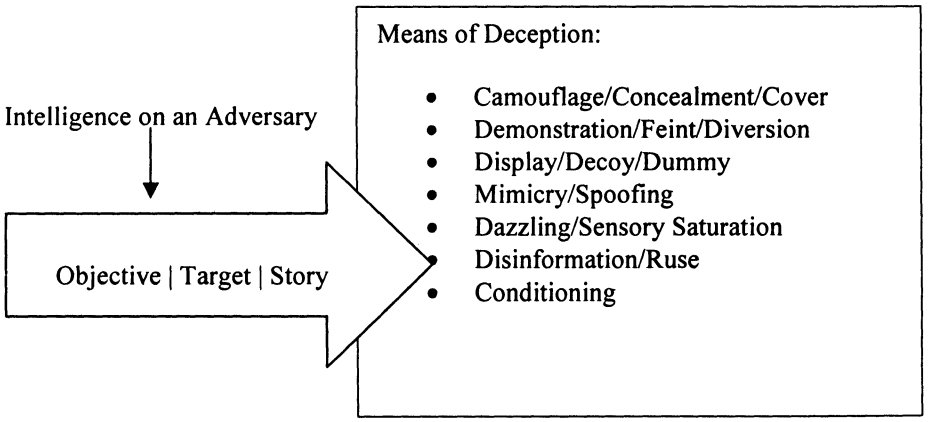

Figure 2: Deception Planning Process (Gerwehr and Glenn, 2003, p.26)

The starting point for implementing such an approach with a networked computing infrastructure would be to outline the targets that an organisation may aim to achieve. There are a number of different targets that an organisation may intend to achieve including the defence of its infrastructure against intruders etc. Intelligence and

information gathering about current attacks and adversaries would also play an important role in informing the actions against further potential attackers.

The following deception techniques could be considered deployed in order to tackle an attacker with intentions of penetrating an organisation's infrastructure by defeating its network defence:

- Concealment or hiding
- Camouflage (hiding movements from the intruder by artificial means)
- False and planted information (misinforming)
- Displays ("techniques to make the enemy see what is not actually there")
- Ruses ("tricks, such as displays that use enemy equipment and procedures")
- Insight ("deceiving the attacker by out thinking him")

It is likely that in the near future, deception implemented through hi-tech means will play an increasing role in both IO and computer network operations. It may be possible to deploy a combination of these techniques in a systematic manner, so that an attacker may be directed through a series of deceptions into deceived states. We intend to study further on how such techniques can be a part of a standard administration task-list, as discussed above and provide more means for information assurance, on top of the traditional, existing information security armoury.

4. Conclusions

Deception is an essential component of military tactics and is becoming an integral part of any successful IO campaign. The significance of PYSOPS in IO, as witnessed in the first Iraq war, reflects the importance of deploying the appropriate deceptive techniques in order to enhance the operation. As the contemporary battlefield, be it conventional warfare or industrial espionage, becomes more and more digitised and interconnected, there is the need to consider effective deceptive techniques as part of network defences and an information security countermeasure.

Although, a number of information and computer systems security related frameworks are available, organisations do not have enough guidance in the field of CND and their infrastructure protection with respect to the deployment of deception. There are no methodologies available that would enable an organisation to employ strategic deception in order to increase the security of its networks. Deployment of deception as CND is still in its infancy. Further research could be directed as to improve defences through the use of deception proactively against a target such as an intruder aiming to penetrate the network of an organisation, by compromising its defences.

References

Armistead, L. (2004). Information Operations, Brassey's Inc., ISBN 157488699-1.

Ashenden, D. & Jones, A. (2003). Re-Interpreting Information Operations for the Private Sector, 2nd European Conference of Information Warfare.

Busuttil, T. & Warren, M. (2003). A Review of critical Information Infrastructure Protection within IT Security Guidelines, 4th Australian Information warfare and IT security Conference 2003.

Campen, A. D. and D. H. Dearth (1998). Cyberwar 2.0: Myths, Mysteries and Reality. Fairfax, Virginia, AFCEA International Press

Cohen, F & Lambert, D. (2001). A Framework for Deception, [online], http://all.net/journal/deception/Framework/Framework.html

Cohen, F. (1998). "A Note on the Role of Deception in Information Protection." Computers & Security 18 November 1998 17(6): 483-506

Computer Security Institute (2005) *Computer Crime and Security Survey*, USA.

Ernst & Young (2005) *Annual Global Information Security Survey*, downloadable from www.ey.com/security

Friedman, A. H. (2003). No-Fly Zone Warning Leaflets to Iraq, [online], http://www.psywarrior.com/IraqNoFlyZone.html

Gerwehr, S. & Glenn, R. W. (2003). Unweaving the Web: Deception and Adoption in Future Urban Operations, Rand, Santa Monica.

Gupta, N. (2003). Improving the Effectiveness of Deceptive Honeynets through an Empirical Learning Approach, [online], http://www.infosecwriters.com/text_resources/pdf/Gupta_Honeynets.pdf

Hildreth, A. S. (2001). CRS Report for Congress: Cyberwarfare, [online], http://www.fas.org/irp/crs/RL30735.pdf

IWS. (2004). Definition of Psychological Operations, [online], http://www.iwar.org.uk/psyops/

JCS. (1996). Joint Doctrine for Military Deception, Joint Pub 3-58, [online], http://www.fas.org/irp/doddir/dod/jp3_58.pdf

JCS. (2003), Doctrine for Joint Psychological Operations: Overview, Joint Publication 3-53, [online], http://www.iwar.org.uk/psyops/resources/doctrine/psyop-jp-3-53.pdf

Lambert, D. (1987). A Cognitive Model for Exposition of Human Deception and Counterdeception), [online], http://jps.lanl.gov/vol1_iss1/5-Cognitive_Model_of_Deception.pdf

Mitnick, K. D. & Simon, W. L. (2002). The Art of Deception. Indianapolis, USA, Wiley Publishing

QinetiQ. (2003). Information Operations (IO), [online], http://www.qinetiq.com/home_enterprise_security/datasheet_index.Par.0002.File.pdf

Rue, L. (1994). By The Grace ff Guile: The Role of Deception in Natural History and Human Affairs. New York: Oxford University Press.

Seymour, J.L. (1996). Deception in Warfare, Compiled on-line by Janet L. Seymour, Bibliographer,
Air University Library Maxwell AFB, AL
http://www.au.af.mil/au/aul/bibs/decwar/dwtoc.htm

Spitzner, L. (2003). Honeypots - Tracking Hackers. Boston: Pearson Education Inc.

Sun, T, (1983). The Art of War, Translated by James Clavell, Dell Publishing, New York, NY.

Sykes, J. B. (1981). The Concise Oxford Dictionary, Clarendon Press

Tryfonas, T. and Kiountouzis, E. (2003), "Perceptions of security contributing to the implementation of secure IS", in Gritzalis, D. et al. (Eds.), Security and Privacy in the Age of Uncertainty, IFIP/SEC'03, Kluwer Academic Publishers, pp. 313-324.

Vidalis, S. (2004). Security Analysis of Micro-payment Systems. Computing Research: Technical Reports, School of Computing. Pontypridd, University of Glamorgan, [online] http://www.glam.ac.uk/socschool/research/publications/technical/CS-04-02.pdf

VMware (2006). Virtual machine for web browsing (browser appliance), [online], http://www.vmware.com/vmtn/appliances/browserapp.html.

Waltz, E. (1998). Information Warfare. Norwood, USA, Artech House

A New Approach to Understanding Information Assurance

Andrew Blyth[1], Colin Williams[2], Ian Bryant[3] and Harvey Mattinson[3]

[1] Information Security Research Group (ISRG), Dvision of Electronics and Computer Systems Engineering, Faculty of Advanced Technology, University of Glamorgan, Pontypridd, RCT, CF37 1DL.
[2] SBL Ltd, East Moor House, Green Park Business Centre, Goos Lane, Sutton on the Forest, York, UK
[3] Central Sponsor for Information Assurance (CSIA), Cabinet Office, HMG, UK

Abstract: The growth of technologies such as ubiquitous and the mobile computing has resulted in the need for a rethinking of the security paradigm. Over the past forty years technology has made fast steps forward, yet most organisations still view security in terms of Confidentiality, Integrity and Availability (CIA). This model of security has expanded to include Non-Repudiation and Authentication. However this thinking fails to address the social, ethical and business requirements that the modern use of computing has generated. Today computing devices are integrated into every facet of business with the result that security technologies have struggled to keep pace with the rate of change. In this paper we will argue that the currently view that most organisations/stakeholders have of security is out-of-date, or in some cases wrong, and that the new view of security needs to be rooted in business impact and business function.

1 Introduction

The growth of technologies related to remote/distance working has lead to the creation of ubiquitous computing and the GRID. GRID and ubiquitous computing function by distributing the processes and storage capacity across a network. This move towards distributed computing has pushed organisations towards the use of shared resources and shared infrastructure. This drive towards co-operative working and resource/infrastructure sharing has resulted in the need to re-think and re-assess the meaning of terms like information assurance, threat and risk management.

2 The Meaning of Security

Before the advent of the personal computer if you wanted to use a computer then you had to make use of a main-frame. These were large computers that where

housed in large computer rooms, and costs millions of dollars. The rainbow book series was a series of books that came out of the US Department of Defense. The Orange book attempted to provide a semantic interpretation of security. It achieved this through the imposition of an ontological framework that allows us to structure and formally represented our understanding of security. This ontological framework views security from a technical/mathematical perspective and lead to the creation of the Bell-LaPadula module of security [5]. Later standards such as ITSEC and Common-Criteria have moved towards a more function descriptive view of security that is cognizant of growth of personal computing devices. While other standards such as BS7799 and ISO-27001 [2] have attempted to approach security from a business perspective. However all of these standards start from an assumption that the stakeholder who owns the security problem is fully aware of what their security requirements are, and thus is full able to articulate them. In this paper we will present a new meaning of security based upon the concept of business impact upon a set of seven assurance requirements. The term business impact is defined as follows:

> *"The result of an information security incident on business functions and the effect that a business interruption might have upon them."*

3 Understanding Security Requirements

The UK Government Strategy for Information Assurance has a goal of producing a "risk assessment/management approach to define security processes and requirements which, in turn would lead to requirements for Products and Services". The GIPSI (General Information Assurance Products and Services Initiative) is a working group which is responsible for overseeing a number of initiatives to achieve this goal.

Within GIPSI a programme of work was created to investigate ways to standardise the expression of business requirements for both functionality and assurance in a clear, concise and meaningful way. The first stage of this work was a survey to elicit the understanding of the scale to which people operating in IA in organisations understood what IA means and what an IA requirement is.

Between February 2006 and March 2006 9,252 questionnaires were dispatched to a broad range of public sector contacts. The majority of these questionnaires were sent via e-mail. Approximately 1,000 follow up telephone calls were made in an attempt to facilitate responses. Of the 9,252 questionnaires sent; 77, or less than one per cent, were returned.

The extremely low rate of return for the questionnaire is highly instructive, as are the reasons given for non return which were obtained during the telephone activity. Some respondents were uncomfortable about providing information of this nature to a commercial third party; however a significant number were uncomfortable about providing information of this nature at all.

It is not possible to publish the returns themselves in this paper. Nonetheless, the returns provide some empirical validation of the following observations:

1. IA professionals lack understanding of the businesses they practice within.

2. Business managers lack an understanding of IA.

3. IA professionals, IT professionals and business managers each talk a different language to the other. Moreover, within each of these stakeholder communities there are multiple means of expression which tend to impede rather than enable common understanding.

4. That there is a difference between IA requirements, an IT/IS requirement and a business requirement is not commonly understood.

5. There is an immature understanding of IA as distinct from more traditional notions of security.

6. Risk management remains an ambition, even a distant aspiration, for some.

7. Some areas of the public sector have yet to match pace with the requirements of the Information Age.

8. There is a pervasive inability to specify a requirement (of any kind) in functional terms. Rather, there is a deep rooted predisposition to articulate in terms of specific product or, at best, technology types. This, or rather the causes of this, may bear further analysis as an a priori root cause explanation for the failure of some public sector IT projects.

All of the above illustrates that as IT has changed, so people's perception of security and what it means has failed to evolve. Consequently what is required is a new definition of security. This definition must focus upon the fact that IT has become and integral part of doing business and that information assurance has become a key driver to commercial success. Thus what is now required is a model that links business impact with security.

4 Business Impact versus Security

We define six levels of business impact numbered 0-5. Each level has a precise meaning.

Level	Business Impact Level (BIL)
0	Negligible
1	Very Low
2	Low

3	Medium
4	Medium-High
5	High/Very High

For each level we define seven assurance requirements, and these are:

Requirements	Assurance Requirements
0	Product Assurance
1	Service Assurance
2	Systems Assurance
3	System Configuration Test
4	Compliance Process
5	Crypto Assurance
6	Protective Making

The combination of these two dimensions can then be combined to produce a simple but elegant two dimensional matrix. The role of this matrix is to allow us to cross reference business impact against assurance requirements. So for a given information assurance requirement, there are seven levels of business impact. At each level of business impact we can define the best practice to mitigate that impact level.

4.1 Product Assurance

Product assurance is concerned with the level of functionality that a given product may contain. Thus product assurance can be seen as the ability of product to function within given performance limits under specified operational conditions over its intended operating life and can include concepts such as usability, reliability and maintainability. Typical ways for measuring product assurance include a functional test against the function requirements specification (FRS).

Business Impact level	Product Assurance
0	Common Best Practice (CBP)
1	CSIA Claims Test (CCT) Mark
2	CSIA Claims Test (CCT) Mark (EAL 1/2)
3	Common Criteria 2-3
4	Common Criteria 2-4
5/6	Common Criteria > EAL 4

4.2 Service Assurance

Service assurance is concerned with the level of functionality that a given service may contain. In particular the level of confidence that the service is free from vulnerabilities, either intentionally designed into the service or accidentally

inserted at anytime during its life cycle and that the service functions in the intended manner

Business Impact level	Service Assurance
0	Common Best Practice (CBP)
1	CSIA Claims Test (CCT) Mark
2	CSIA Claims Test (CCT) Mark (EAL 1/2)
3	Future Assurance Model
4	Future Assurance Model
5/6	Future Assurance Model

4.3 System Assurance

System assurance is concerned with the level of functionality that a given system may contain. In particular the level of confidence that the system is free from vulnerabilities, either intentionally designed into the system or accidentally inserted at anytime during its life cycle and that the system functions in the intended manner.

Business Impact level	System Assurance
0	Common Best Practice (CBP)
1	CSIA Claims Test (CCT) Mark
2	CSIA Claims Test (CCT) Mark (EAL 1/2)
3	Low Tailored Assurance
4	High Tailored Assurance
5/6	Common Criteria (CC) > EAL 4

4.4 System Configuration Test

System Configuration test is concerned with the provenance of a level of functionality that a given system configuration may contain. In particular, the testing of this functionality to prove that system configuration precisely supports it and is free from error and omissions that could be used to breach the security of the system.

There is plenty of evidence to support the assertion that failure to correctly configure and manage configuration can result in vulnerabilities being introduced into the system [4].

Business Impact level	System Configuration Test
0	Common Best Practice (CBP)

1	Common Best Practice
2	Common Best Practice
3	Professionally Certified Penetration Test (PCPT)
4	PCPT + Vulnerability Test (VT)
5/6	PCPT + VT + Managed IDS

4.5 Compliance Process

Compliance Process is concerned with the level of functionality/assurance that a given Compliance Process may contain. Its purpose is to measure the level to which a process is adhered to.

Business Impact level	Compliance Process
0	Internal Audit
1	ISO 27001 Audit
2	ISO 27001 Audit
3	Accreditation + ISO 27001
4	Accreditation + ISO 27001
5/6	Accreditation + ISO 27001

4.6 Crypto Assurance

Crypto Assurance is concerned with the level of assurance that a given crypto process/product may contain. In essence it is the level to which a measurement can be made of confidence that we can have in the functionality of a cryptographic process/product.

Business Impact level	Crypto Assurance
0	Common Best Practice (CBP)
1	CCT Mark
2	FIPS 140-2 + CCT Mark
3	CESG Assisted Products Scheme (CAPS) Baseline
4	CESG Assisted Products Scheme (CAPS) Enhanced
5/6	CESG Assisted Products Scheme (CAPS) High

4.7 Protective Marking

Protective Marking is concerned with the level of sensitivity that a given asset may contain. It is used to define the procedures that should be used to access and handle such information.

Business Impact level	Protective Marking
0	Unclassified - Public
1	Due Care
2	Unclassified but Sensitive
3	Restricted
4	Confidential
5/6	Secret and Top Secret

5 Conclusion

In conclusion we would say that this new model of security directly address the need for information assurance to be linked to need and expectations of a business. Evidence has shown that information assurance can not be bolted onto a system or service as an after thought and that it is integral to the functioning of any business process. Loss of information assurance can in the commercial world be disastrous. Consequently various types of information assurance requirements to be linked to various types of business requirements

6 Acknowledgements

This research has been supported by the UK Central Sponsor for Information Assurance (CSIA) and SBL Ltd. We would also like to acknowledge all of the members of the GIPSI committee would have helped in this research.

7 References

[1] ISO 13335, Information technology -- Security techniques -- Management of information and communications technology security -- Part 1: Concepts and models for information and communications technology security management, ISO Standards, 2004

[2] ISO 27001, Information technology -- Security techniques -- Information security management systems – Requirements, ISO Standards, 2005

[3] ISO 17799, Information technology -- Security techniques -- Code of practice for information security management, ISO Press, 2005

[4] Peter G. Newmann, Computer Related Risks, ACM Press, 1995.

[5] D. Elliott Bell and Leonard J LaPadula, Secure Computer Systems: Mathematical Foundations, MITRE Corporation, 1973.

Robust Public Key Cryptography – A New Cryptosystem Surviving Private Key Compromise[1]

Cheman Shaik
A.H.Industrial Consultants, Information Technology Group
P.O.Box 56565, Riyadh 11564, Saudi Arabia
cheman_shaik@rediffmail.com

1. Introduction

A weakness of the present-day public key cryptosystems is that these cryptosystems do not survive private-key compromise attacks resulting from an internal breach of trust. In a competitive business environment, private key compromise is a common incident that voids the strength of public key cryptosystems such as RSA and ECC. Bribing corporate employees to disclose their secret keys and inadvertently disclosing secret information are among a plethora of practical attacks that occur at the implementation level. Once a breach of trust takes place and subsequently the private key is revealed, any public key cryptosystem fails to secure electronic data in Internet communications. The revealed key may be used by an attacker to decipher the intercepted data at an intermediary router. This weakness of public key cryptography calls for an additional security measure that enables encryptions to survive private key compromise attacks.

2. Robust Public Key Cryptography

Robust Public Key Cryptography is a new technique that overcomes the shortcoming of public key cryptography discussed above. In this technique, both the keys, public and private, contain two exponents. Using the two exponents of the encrypting key, the original message is encrypted into two different ciphertexts. The two ciphertexts are routed to their destination through two different paths via the Internet. Upon receiving the two ciphertexts, certain exponential and multiplication modular operations are performed to decrypt the message. Double-path routing of ciphertexts is achieved through *source routing,* an IP option that allows the originator of a packet to specify what path it will take to its destination. Before delivering the ciphertexts, two returnable test

---

1 An early draft of this paper was accepted as a poster display and presented at EC2ND 2005; it has since then earned a U.S. Patent.

packets may be delivered to the same destination in order to know two different paths to the destination. Fig.1 shows typical source routing of two ciphertexts.

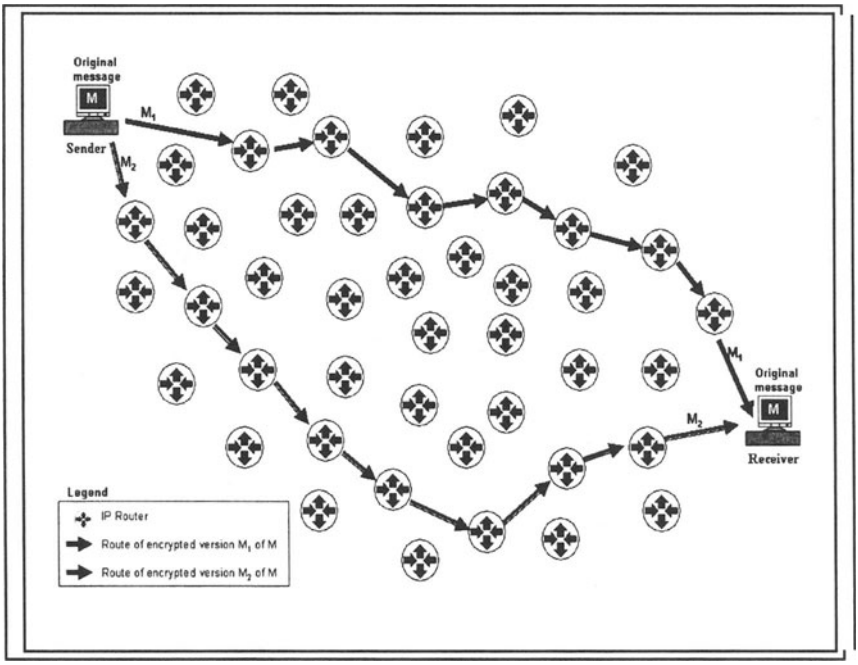

Fig. 1 Source Routing of Ciphertexts

It is practically infeasible to capture both the ciphertexts routed through two different paths. An attacker needs to have access to the wire that the communication is going across in order to eavesdrop. If an attacker occasionally happens to be on such a path, he has access to one of the ciphertexts while it is not possible for him to capture the other following a different route. With a single ciphertext captured at a router an eavesdropper's decryption produces junk. Decryption will not be complete unless both the ciphertexts of the original message are collected and substituted in the appropriate formula.

The new cryptosystem provides two layers of security over Internet communications. The first is the conventional RSA security layer that comes from the computational effort involved in factorizing the key modulus into two primes while the second one comes from double-path routing of the ciphertexts of the original message. This additional layer

can be broken only when the attacker captures both the ciphertexts, which is practically infeasible in real networking conditions on the Internet.

When the private key is kept confidential and the attacker lacks one ciphertext, both layers together protect the confidentiality of Internet communications. In case the private key is compromised and the attacker lacks one ciphertext, the second layer provides security. In the third case where the private key is kept confidential and both the ciphertexts are captured, the conventional RSA security layer still protects the confidentiality of communications, making the cryptosystem at least as secured as the RSA. The private key of the *Robust Public Key Cryptosystem* is called *Robust Private Key* as it is invulnerable to compromise attacks.

Robust Public Key cryptography should not be confused with society or group oriented cryptography introduced by Yvo Desmedt [1].

3. Blind Key Algorithm

This section presents an implementing algorithm called *Blind Key Algorithm* that performs blind encryptions. A blinding number is selected randomly and discarded after completing the encryption task.

The algorithm is based on the following two equations governing the relations among the key exponents:

$$e_1.d_1 + e_2.d_2 = k_1.\varphi + 1. \quad \textbf{(1)}$$

$$d_1 + d_2 = k_2.\varphi. \quad \textbf{(2)}$$

p and q are two primes; n = p.q ; φ = Euler Totient Function of n =(p-1).(q-1) ;
$\{d_1, d_2\}$ - Robust Private Key ; $\{e_1, e_2, n\}$ - Public Key ; k_1 , k_2 - arbitrary constants ;
p, q and φ are discarded once keys are computed.

Once φ and n values are established by choosing the primes p and q, computation of keys can be initiated by selecting an arbitrary integer k_2 in eqn.2. d_1 and d_2 are selected such that eqn.2 is satisfied. By arbitrarily selecting e_1, the remaining two integers, e_2 and k_1, can be fixed by the Euclidean algorithm.

Before encrypting a message a blind encryption key is formed by adding a random integer t to each exponent of the key. Therefore, a blind key (e_1+t, e_2+t, n) is used for encryption. $t < \varphi$ is a random number selected on the sender's machine before encryption and discarded once encryption is complete with no trace for later reference. Encryption and decryption are performed according to the following steps:

The sender encrypts his message $M \in \{0,\ldots,n-1\}$ into two ciphertexts M_1 and M_2 by performing exponential modular operations on M as follows:

$$M_1 = M^{e_1+t} \bmod n. \quad (3)$$

$$M_2 = M^{e_2+t} \bmod n. \quad (4)$$

M_1 and M_2 are delivered to the receiver.

Similarly, the receiver computes N_1 and N_2 as follows:

$$N_1 = M_1^{d_1} \bmod n. \quad (5)$$

$$N_2 = M_2^{d_2} \bmod n. \quad (6)$$

Next, the receiver performs multiplicative modular operations on N_1 and N_2 to compute N as follows:

$$N = N_1.N_2 \bmod n. \quad (7)$$

This ends the cryptographic process on both sides. At the end of the above process, the resulting value N is equal to M, the original message.

4. Proving N=M

The equality N = M may be proved as follows:

From eqn.7 $N = N_1.N_2 \bmod n.$

Substituting for N_1 and N_2 in the above,

$$N = [(M_1^{d_1} \bmod n).(M_2^{d_2} \bmod n)] \bmod n. \quad (8)$$

According to modular arithmetic properties the above equation simplifies to

$$N = [(M_1^{d_1}.M_2^{d_2})] \bmod n. \quad (9)$$

Substituting for M_1 and M_2 in the above,

$$N = [(M^{e_1+t} \bmod n)^{d_1}.(M^{e_2+t} \bmod n)^{d_2}] \bmod n. \quad (10)$$

Again, according to modular arithmetic properties the above equation simplifies to

$$N = [M^{(e_1+t)d_1} . M^{(e_2+t)d_2}] \bmod n. \qquad (11)$$

Rearranging the exponents in the above,

$$N = [M^{e_1.d_1+ e_2.d_2 + t(d_1+d_2)}] \bmod n. \qquad (12)$$

Substituting eqn.1 and eqn.2 in the above,

$$N = (M^{k_1.\varphi+1 + t.k_2.\varphi}) \bmod n. \qquad (13)$$

Simplifying the exponent in the above,

$$N = (M^{k_3.\varphi+1}) \bmod n. \qquad (14)$$

k_3 in the above equation may be expressed as $k_3 = k_1 + t.k_2$.
According to Euler's theorem in number theory, $M^{k.\varphi+1} \bmod n = M$ for any M, k, and n when M and n are relatively prime. Hence, N = M.

Interestingly, it can be seen from the above proof that the effect of blinding the original public key with a random number ***t*** becomes void when both the ciphertexts are substituted in the decryption formula. David Chaum also employed the concept of blinding for providing anonymity to spenders of his electronic cash [2, 3].

5. Breaking the Security

Assuming that the attacker knows only one ciphertext, say M_2, and the private key as it is compromised already, the original plaintext M can be computed if *t* is known. Since *t* is randomly selected and discarded after encryption, it is not possible to recover the original plaintext. When at least one ciphertext could not be captured, the only way to break the security is to attempt a brute force decipher attack with all possible *t* values for which the attacker has to scan through the entire key space.

A brute force attacker needs to compute the multiplicate inverse e_t of (e_2+t) for every assumed *t*, which involves many iterations for each run of Euclid's algorithm [4]. Since the order of φ is same as n, the problem of computing all the multiplicative inverses is at least $\sqrt{n}$ times harder than the RSA problem.

Further, the attacker is required to compute $M_2^e \bmod n$ with each multiplicative inverse corresponding to the assumed ***t***. This makes the

problem even more harder by a factor of $b(b+1)/4$ where b is the size of key modulus in bits. Hence, the ultimate security factor over the RSA problem is $[b(b+1).\sqrt{n}]/4$.

Another advantage of the *Robust Public Key Cryptosystem* is that every message communicated in this method is individually secured, meaning even with a compromised private key it is necessary to make the same computational effort to recover every message, thereby preventing single-hour bankruptcies of e-commerce merchants.

6. Conclusion

A new public key cryptosystem called *Robust Public Key Cryptography* is introduced and an implementing algorithm is presented. An advantage over the public key cryptosystems, survival against private key compromise, is emphasized. It is shown that the implementing algorithm is too hard to break when only one ciphertext could be captured, even after the private key is revealed. Private key confidentiality maintained by the key owner additionally imparts the conventional RSA security to communications. Hence, it is recommended that the users of *Robust Public Key Cryptography* do not to reveal the private key intentionally though double-path routing protects confidentiality of their messages. The technique may be best utilized by exploiting both layers of security. The second layer of double-path routing serves as a shield against any internal trust breaching elements.

References

[1] Desmedt, Yvo. Society and Group Oriented Cryptography: A new concept. *Advances In Cryptology – Crypto '87*, pages 120-127.

[2] D.Chaum. Blind Signatures for Untraceable Payments. *Proceedings of Crypto '82*, pages 199-203.

[3] D.Chaum, A.Fiat, M.Naor. Untraceable Electronic Cash. *Proceedings of Crypto '88,* pages 319-327.

[4] William Stallings. Cryptography And Network Security, Principles and Practice, pages 223-225. Prentice Hall, Second Edition.

The Second European Conference on Computer Network Defence, inc. the First Annual Workshop on Digital Forensics & Incident Analysis

Section II: Digital Forensics & Incident Analysis

Review of Forensic Tools for Smartphones

Hamid Jahankhani, Amir Azam
University of East London
School of Computing and Technology
UK
hamid.jahankhani@uel.ac.uk

Abstract: The technological capability of mobile devices in particular Smartphones makes their use of value to the criminal community as a data terminal in the facilitation of organised crime or terrorism. The effective targeting of these devices from criminal and security intelligence perspectives and subsequent detailed forensic examination of the targeted device will significantly enhance the evidence available to the law enforcement community. When phone devices are involved in crimes, forensic examiners require tools that allow the proper retrieval and prompt examination of information present on these devices. Smartphones that are compliant to Global System for Mobile Communication (GSM) standards, will maintains their identity and user's personal information on Subscriber Identity Module (SIM). Beside SIM cards, substantial amount of information is stored on device's internal memory and external memory modules. The aim of this paper is to give an overview of the currently available forensic software tools that are developed to carry out forensic investigation of mobile devices and point to current weaknesses within this process.

Keywords: Smartphones, digital forensics tools, PDA, SIM, USIM

1. Introduction

Internet ready mobile devices are hitting the market every day, with increasing processing power and storage media capabilities. As innovative technology and customer demand cause the accumulation of all new device capabilities, the 3G multimedia device or 3G personal companion will become the sought after all-in-one mobile Internet tool for the middle of the decade. Currently there are stand-alone single wireless gadgets that will meet all the needs of mobile user but not limited to these as follows:

- **Smart phones/WAP (Wireless Application Protocol) phones:** These devices provide web browsing and some enhance features by using WAP protocols based new operating systems, and also synchronize with other devices (like desktops and mobile phones).These phones evolve to a talkative PDA (smart phones).

- **Talkative PDA (Personal Digital Assistant)/ Smart phones:** Although there is room for more enhancements and integrations, today you can purchase a PDA that has mobile voice communications protocols. Besides their PIM applications e.g. calendars, address books, and other organizing features, these devices are thin and lightweight; color screens, and are quickly gaining computer strength

due to low power chip designs, screen miniaturization, and evolving operating systems. As they grow in computing power while maintaining their hand-held form factor, they will continue to distinguish themselves from 3G laptops as less expensive, less powerful solutions. Examples are numerous with Palm, Casio, HP, and others leading the pack.

- **Handheld Internet 3G laptops:** Laptops today have internal WiFi, network interface card (NIC), modems and Personal Computer Memory Card International Association (PCMCIA) cards that enable wireless communications. They continue to get smaller, lighter, and with more powerful computing combined with the bandwidth offered by 3G, these powerful, portable computers will thrive with the custom graphics, two-way video conferences, and large file transfers for the future.

- **Handheld Internet 3G Web Tablet:** These devices offer portable Internet access by plugging into power and gaining limited mobility via a wireless connection. As low-cost, lightweight, thin Internet appliances the size of magazines, these devices offer e-mail, robust Internet access, and web browsing. Eventually, they will gain both full mobile access and synchronization with other devices via more powerful 3G spectrum.

- **3G multimedia device (personal companion):** 3G will solve the issues of slow connections based devices which cause unsteady video images because compression techniques cannot overcome the need for speed and capacity.

Smartphone has an optimized platform for applications that allow real time or off line access to information with minimal input from the user. An example might be a scheduling application that allows a user to quickly look up details of their appointments for the day and download the required background information, (Microsoft mobile solutions, 2003).
Unlike many traditional mobile phones, smartphones allow individual users to install, configure, and run applications of their choice. A smart phone allows the user to set the setting to suit them i.e. download mp3 music or making it multi-purpose remote control. Most standard software offers only limited choices for re-configuration, forcing you to adapt to the way it is set up for example on a standard phone, whether or not you like the built-in calendar application, that is the only your option, (Electronics, 2006).
Within the UK, the technical ability to interrogate telephone systems did not start to materialize until the late 1960s and early 1970s with the gradual role-out of the Subscriber Trunk Dialling (STD) system. Phone forensics at that time simply did not exist. During the mid 1980s, the legal precedents caused by the Police and Criminal Evidence Act (PACE 1984) with it's associated requirement for investigating officers to be legally accountable for their actions and the veracity of any claims required a totally different approach to the presentation of evidence. At this time the embryonic analogue mobile phone were being deployed in the US and key European cities. Pioneered in the UK by British Telecommunications (BT) and Securicor as "Cellnet" and Racal Vodaphone as "Vodaphone", they relied initially on FM high-band UHF

analogue handsets whose functionality was nothing more than full duplex analogue trunk radio. Forensic examination of the handsets was limited to the traditional physical aspects and networked based information allowing only for call origination, destination, timing, cell identification and user identification. The analogue handsets were however notoriously vulnerable to interception (their signals were unencrypted) and cloning as security protocols was similarly "in clear". Nonetheless, the gathering of mobile phone evidence as part of a prosecution was an accepted act but was still treated in the same vein as traditional landline evidence, albeit with intercept traffic being utilised as intelligence material; analysis being limited to transcript of voice content.

The introduction of the TDMA based digital GSM system in the early 1990s (based on the 1988 GSM standard promulgated by the European Telecommunications Standards Institute, (ETSI, 2006), brought a functional revolution with three global variants being developed; in the UK this was to replace the TACS FM system. Statistically, there are now over 342 million mobile subscribers in Europe, with UK market share of 54.3 million subscribers, i.e., nearly every person in the UK over the age of 10 has a mobile phone, (Focus Group, 2006).

Given the number of mobile devices and their acceptance as a means of communication within any social grouping, it is clear that their use by the criminal fraternity is a given.

The focus of this paper is on evaluation and comparison of different mobile forensics tools currently available.

2. Generic Smart Phone overview

In general, as shown in figure 1 the most generic architecture format of smartphones consist of a microprocessor, Read Only Memory (ROM), Random Access Memory (RAM), a variety of hardware keys and interfaces, communication protocols and a touch sensitive liquid crystal display.

The Operating System (OS) of these devices is held in ROM. several varieties of ROM are used, including Flash ROM, which can be erased and reprogrammed electronically with OS updates or an entirely different OS. RAM, which normally contains user data, is kept active by batteries whose failure or exhaustion causes all information to be lost.

The latest smart phones come with extra memory capacity modules e.g. Compact Flash (CF), Secure Digital (SD), MultiMedia Card (MMC), Micro Drives, and peripherals, such as a digital camera or Bluetooth or WiFi built in and many more to come.

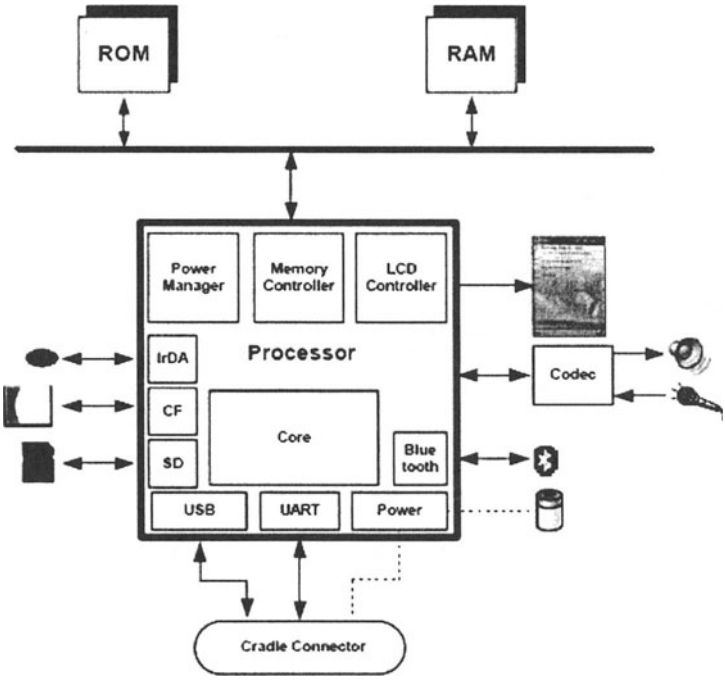

Figure 1: Generic Hardware Overview, Source: (Ayers , 2004)

2.1. Removable media

Removable media extends the storage capacity of a smart phone, allowing individuals to store additional information beyond the device's built-in capacity. Removable media is non-volatile storage, able to retain recorded data when removed from a device.

The storage capacities of memory cards range from MegaBytes (MB) to GigaBytes (GB) and come in sizes literally as small as a thumbnail. As technological advances continue, such media is expected to become smaller and offer greater storage densities. Fortunately, such media is normally formatted with a conventional file system e.g., FAT16, FAT32 (File Allocation Table) and can be treated similarly to a disk drive, imaged and analyzed using a conventional forensic tool with a compatible media adapter that supports an Integrated Development Environment (IDE) interface. Such adapters can use with a write blocker to ensure that the contents remain unaltered. Below is a brief overview of several commonly available types of memory cards used with phones, (Blue mug, 2006).

This list does not include every media type available; rather, it is intended to show the variety of media types that an analyst may come across.

Media Type	Reader	Capacity	Comments
Flash/Jump drive	USB interface	16 MB – 6 GB	Also known as thumb drive because of their size
Compact Flash card	PCMCIA adapter or memory card reader	16 MB – 6 GB	Type I cards measure 43 x 36 x 3.3 mm; Type II cards measure 43 x 36 x 5 mm
Micro drive	PCMCIA adapter or memory card reader	340 MB – 8 GB	Same interface and form factor as Compact Flash Type II cards
MultiMedia Card (MMC)	PCMCIA adapter or memory card reader	16 MB – 2 GB	Measure 24 x 32 x 1.4 mm
Secure Digital (SD) Card or mini SD Card	PCMCIA adapter or memory card reader	32 MB – 2 GB	Compliant with Secure Digital Music Initiative (SDMI) requirements; provides built-in data encryption of file contents; similar in form factor to MMCs
Memory Stick	PCMCIA adapter or memory card reader	16 MB – 2 GB	Includes Memory Stick (50 x 21.5 x 2.8 mm), Memory Stick Duo (31 x 20 x 1.6 mm), Memory Stick PRO, Memory Stick PRO Duo; some are compliant with SDMI requirements and provide built-in encryption of file contents
Smart Media Card	PCMCIA adapter or memory card reader	8 MB – 2 GB	Measure 37 x 45 x 0.76 mm
xD-Picture Card	PCMCIA adapter or xD-Picture card reader	16 MB – 1 GB	Currently used only in Fuji film and Olympus digital cameras; measure 20 x 25 x 1.7 mm

Table 1: Memory Media Comparison Source, (Chevalier, 2005)

2.2. The Subscriber Identity Module (SIM) Types

In Europe GSM is stronger market and GSM based phones works on SIM cards. SIMs originated with a set of specifications developed by CEPT (Conference of European Posts and Telecommunications) and continued by ETSI (the European Telecommunications Standards Institute) for GSM networks.
The Subscriber Identity Module (SIM) is an essential component of a GSM and Universal Mobile Telecommunications System (UMTS); which is use to uniquely identify the subscriber or in other words owner or user of that mobile device to GSM network. It also holds other user-related information. It stores network state information such as its current Location Area Identity (LAI).
When the handset is turn off and back on, it will take data off the SIM card and search for the LAI that was registered last time phone was on. This saves time by not searching the whole list of frequencies that the phone normally would have to do in fresh location to setup communication with broadcasting its International Circuit Card ID (ICCID) [49] (globally this ID uniquely identifies each SIM cards).
The SIM therefore is of great importance to the investigator. Since it is just like smart card but smaller in size and can be removed from a phone and read by using a specialized SIM card reader with help of standard-size smart card adapter and a conventional smart card reader.
The SIM contains following but not limited to, (Willassen, 2005):

- **Basic Data**: subscriber information Customer Identification (IMSI), including PIM data, List of phone numbers dialed by the user

- **Location Data**: service provider identification; data regarding where the device was last used for a particular service and any "forbidden" networks it might have encountered

- **Enhanced Messaging Service (EMS) Data**: long text messages (over 160 characters) and messages with combination of simple media such as melodies (ringtones), pictures, sounds, animations, modified text and standard text as an integrated message; including deleted messages that are impossible to view on phones

- **Foreign Language Data**: SMS messages and PIM data written in a foreign language

- **Other**: In some cases, information about countries visited by the user

In order to obtain evidence from any smart card, the investigator needs the correct access code(s); Personal Identification Numbers (PIN) or Chip Holder Verification (CHV) to access full contents of SIM cards beside any other device's OS restriction.
In SIM cards, Two PINs exist, sometimes called PIN1 and PIN2 or CHV1 and CHV2. The user can enable or disable these PINs but by default they are disabled.

The SIM only allows a set number of attempts, usually three, to enter the correct PIN before further attempts are blocked. Entering the correct PUK (PIN Unblocking Key) resets the PIN number and the attempt counter. The PUK can obtain from the service provider with seized written notices from forensic unit or the network operator based on the SIM's identity (i.e., its ICCID). If the number of attempts to enter the PUK code exceeds a set limit, normally ten attempts, then that card blocked itself permanently for any further use, (Willassen, 2005).

Most popular and upcoming SIMs is as follows:

USIM: Universal Subscriber Identity Module, (Wikipedia, 2006) is an application (running on smart card) for UMTS mobile telephony, which is use for a 3G mobile phone. It stores user subscriber information, authentication information and plus extra storage (128 kb) space for storing different information on SIM such as enhanced phonebook up to 250 entries with advance field for email, home and business addresses.

ISIM: IP Multimedia Services Identity Module, (Wikipedia, 2006) is an application (running on a smart card) for UMTS mobile phones, which use in a 3G telephone in the IP Multimedia Subsystem (IMS). It contains parameters for identifying and authenticating the user to the IMS. The ISIM application can co-exist with SIM and USIM on the same smart card making it possible to use the same smart card in both GSM networks and earlier releases of UMTS.

W-SIM (Willcom SIM): It has all basic function of regular SIM card and has the core components of the cell phone such as the radio and transmitter that built inside the card. It currently used in the Sharp W-Zero3 Smart phone, and Willcom's TT phone and DD USB cellular device, (Wikipedia, 2006).

R-UIM/ RUIM: Re-Usable Identification Module is a removable smart card for mobile phones made for first time to work on the Code Division Multiple Access (CDMA) network. The RUIM card holds a user's personal information just like SIM card such as name and account number, phone number, phone book, text messages and other settings, (RUIM, 2006).

HC SIM: High Capacity SIM or HD SIM (High Density) SIM and SuperSIM, all of them have same functions as standard SIM card. Moreover, extra massive storage of 16MB to 256MB with integrated increased communication speeds protocols e.g. BIP (Bearer Independent Protocol (BIP), a high-speed interface that opens new channels to the SIM (in addition to the SMS channel) such as General Packet Radio Service (GPRS) and 3G, making it possible to distribute content and applications to the SIM at much higher rate, (Constantinou, 2006).

MSIM: MegaSIM also belongs to high density SIMs family. It comes with flash storage of 64MB to 1GB, independent processing power and high-speed interface like HS-SIMs. MSIM also includes secure downloading of MMS, video clips, full PIM functionality, and personal storage for large databases of any thing user like. It is ideal for smart phones to store very little information on SIM card and take it with them or switch to any other phone instantly, (mSystem, 2006).

Despite of so many different SIM mentioned above, are they secure enough for a user to leave their very personal detail on them and hoping that digital data on this tiny chip card can never be hacked.

2.3. SIM Card Contents:

The evidence on the SIM card is stored in the following files but not limited to only these files:

Phase	Phase ID	1 byte
SST	SIM Service table	5 bytes
ICCID	Integrated Circuit Chip Identifier (Serial Number)	10 bytes
LP	Languages Preferred variable	variable
SPN	Service Provider Name	17 bytes
MSISDN	Mobile Station International Subscriber Directory Number (Subscriber phone number)	variable
AND	Short (Abbreviated) Dial Number	variable
FDN	Fixed Dialing Numbers variable	variable
LND	Last Dialed numbers	variable
EXT1	Dialing Extension 1	variable
EXT2	Dialing Extension 2	variable
GID1	Groups ID level 1	variable
GID2	Groups ID level 2	variable
SMS	Text Messages Services (messages)	n*176 bytes
SMSP	Text Message Services Parameters	variable
SMSS	Text message status	variable
CBMI	Cell broadcast message identifier selection Preferred network messages	variable
PUCT	Per Unit Cost (price per unit charge)	5 bytes
ACM	Accumulated Call Meter (Charge counter)	3 bytes
ACMmax	Accumulated Call Meter maximum (Charge limit)	3 bytes
HPLMNSP	Home Public Land Mobile Network Search Period	variable
PLMNsel	Public Land Mobile Network selector	variable
FPLMN	Forbidden PLMNs	12 bytes
CCP	Capability Configuration Parameter	14 bytes
ACC	Access Control Class	2 bytes
IMSI	International Mobile Subscriber Identity	9 bytes
LOCI	Location Information	11 bytes
BCCH	Broadcast Control Channels	16 bytes
Kc	Ciphering key	variable
AD	Administrative data	variable

Table 2: SIM Content Source, (Mislan, 2006), (3GPP, 1999)

3. Forensic Tools:

Unlike the situation with personal computers, the number and variety of forensic tools for smart phones is considerably limited, but the range of devices over which they operate is also limited due to distinct platforms for a manufacturer's product line (e.g. palm OS, Windows CE and others), a family of operating systems, or a type of hardware architecture.
Some tools provide a full range of acquisition, examination, and reporting functions, (Paraben, 2006), whereas other tools focus mainly on a single function such as SIM forensic, external memory modules (CF, SD, MMC & other) and phone itself, (Ayers, 2004, 2006), Similarly, different tools may be capable of using different interfaces (e.g., IrDA, Bluetooth, serial cable and USB) to acquire device contents. The types of information which a tool can acquire is usually depends on tools specification and vendors hardware and software compatibility. Most commonly available range is as PIM data, logs of calls, messages, email, URLs (Uniform Resource Locator), video, audio, image, and SIM data. In order to retrieve entire data from these phones, we sub-categorized them as follows:

- Handset based Tools
- Operating System Based Tools
- SIM based Tools

3.1 Handset Based Tools:

	Function	Features
pilot-link	Acquisition	• Targets Palm OS phones • Open source non-forensic software • No support for recovering SIM information • Supports only cable interface
Device Seizure	Acquisition, Examination, Reporting	• Targets certain models of GSM, TDMA, CDMA , with Palm OS, Pocket PC, and RIM OS advance handheld devices support • Supports data recovery of internal and external memory • Supports cable, Bluetooth, and IR interfaces
GSM .XRY	Acquisition, Examination, Reporting	• Targets certain models of GSM phones • Supports recovery of internal and external SIM • Supports cable, Bluetooth, and IR interfaces
Oxygen PM (forensic version)	Acquisition, Examination, Reporting	• Targets certain models of GSM phones • Supports only internal SIM acquisition

MOBILed it! Forensic	Acquisition, Examination, Reporting	• Targets certain models of GSM phones • Internal and external SIM support • Supports cable and IR interfaces
BitPIM	Acquisition, Examination	• Targets certain models of CDMA phones • Open source software with write-blocking capabilities • No support for recovering SIM information
TULP 2G	Acquisition, Reporting	• Targets GSM and CDMA phones that use supported protocols to establish connectivity • Internal and external SIM support • Requires PC/SC-compatible smart card reader for external SIM cards • Cable, Bluetooth, and IR interfaces supported

Table 3: Handset Based Tools Comparisons, (Ayers, 2004, 2006)

3.2 OS Based Tools

	Palm OS	Pocket PC	Linux
Device Seizure	Acquisition, Examination, Reporting	Acquisition, Examination, Reporting	-
pilot-link	Only Acquisition	-	-
EnCase	Acquisition, Examination, Reporting	-	Examination, Reporting

Table 4: OS Based Tools Comparisons, (Ayers, 2004, 2006)

3.3 SIM Based Tools

	Function	Features
Device Seizure	Acquisition, Examination, Reporting	• External/ Internal SIM cards (direct / indirect)
USIM Detective	Acquisition, Examination, Reporting	• External SIM cards only (direct)
TULP 2G	Acquisition, Reporting	• Recover information from SIM card, when inserted in handset (No direct SIM support).
GSM .XRY	Acquisition, Examination, Reporting	• Recover information from SIM card, when inserted in handset (No direct SIM support).
Mobiledit! Forensic	Acquisition, Examination, Reporting	• Recover information from SIM card, when inserted in handset (No direct SIM support).
SIMIS	Acquisition, Examination, Reporting	• External SIM cards only (direct)
ForensicSIM	Acquisition, Examination, Reporting	• External SIM cards only (direct) • Produces physical facsimiles of SIM for prosecutor and defense, and as a storage record.
Forensic Card Reader	Acquisition, Reporting	• External SIM cards only (direct)
SIMCon	Acquisition, Examination, Reporting	• External SIM cards only (direct)

Table 5: SIM based Tools Comparisons, (Ayers, 2004, 2006)

3.4. SIM Contents Recovery

Data	Cell Seizure	GSM XRY	Mobile Edit	USIM Detective	SIMIS	Forensic SIM	Device Seizure	SIM Con
IMSI	X	X	X	X	X	X	X	X
ICCID	X	X	X	X	X	X	X	X
SPN	X			X	X	X	X	
USIM	X	X		X	X	X	X	
SST				X	X	X	X	X
LP	X			X	X	X	X	X
ADN	X	X	X	X	X	X	X	X
LND	X	X	X	X	X	X	X	X
SMS	X	X	X	X	X	X	X	X
FPLMN	X			X	X	X	X	X
LAI	X	X		X	X	X	X	X
RAI	X			X	X	X	X	X

Table 6: SIM Content Recovery Comparisons, (Ayers, 2004, 2006)

3.5. Review of flaws and weaknesses

3.5.1. GSM SIM Requirement Bypass Mechanism

One of the major flaw or weakness in GSM phone technology is that their built in mechanism will not allow these devices to perform any further function without a SIM cards. Let us suppose that an investigator do have a smartphone for data retrial but SIM card has been lost or destroyed in the process of acquisition. Now it is just NOT as simple to connect that device with any forensic tool to get a data out of device's memory (usually as regular routine) because it will not go to the next screen until and unless there is a SIM in the device. From forensic point of view we cannot just add any other dummy or test SIM in phone to progress the investigation because device (Totally depends on smartphone's OS and setup) will lose or erase its previous call logs and all other information that may be critical for a case, which usually (by default) will be re-written by contents of new SIM card.
In this type of situation, if tool developer some how can "bypass" the SIM card required screen rule to progress their test, will then make investigator's task much easier, other wise investigator needs to duplicate or fool the device by using look-a-like SIM to the original SIM card, which was last time used in that device. It can be

done in specialized forensic labs but it is time consuming and costly, (Radio Tactics, 2006).

3.5.2 User Login

It is not too hard to implement user based login function in a forensic tool that keeps the tracks of each user actions, when printing the final case report i.e. it should automatically include all the details related to a specific case. It could include simple details like who has accessed this case? How many times? What has been search for?, etc. which could serve the purpose of chain-of-evidence in digital environment otherwise an investigator still have to do a separate report stating all the actions taken on that particular case.

3.5.3. Reports

Reporting method can be improved by including if some how an automatic brief description of steps taken during examination, such as string searches, graphics image searches, recovering erased files, etc., in their report, otherwise the report is too technical and not easy to understandable without help of further expert witness report or investigator explanation.

3.5.4. Viewing and Antivirus

The viewing method is limited to view data as Hex or Text formats, and if you need to view other formats i.e. images, video, etc then other external applications (media player, mp3 player, etc) which are pre-installed with the OS or independently installed on forensic workstation need to be used.
It does not sound forensically right because there is safety risk here, imagine a video file is affected with virus and when it is opened with external application, it can affect the evidence. It is highly recommended to integrate extra viewing functions into the forensic tool to avoid the risk of initiating self destructive virus to wipe the evidence.

3.5.5. Hardware Standardization

When it comes to hardware, which should be used in a forensic process it must be forensically certified or globally standardized compliant. Current forensic tool supports serial type SIM card reader which is not compatible with PC/SC standards where it should be the other way around to fulfil the forensic lab certification requirements.
New smartphones can support number of different external cards with different sizes up to 10GB (one or more modules) or more as mentioned in table 1 and at moment they do support FireWire connection (also known as i.Link or IEEE 1394) for higher data communication to transfer massive data from these memories to other devices.

3.5.6. Timeline

Timeframe or timeline analysis can be useful in determining the number of files that has been accessed, modified, created, etc on digital evidence with relevant to the investigation timeframe. This method will show all hidden files, encrypted files, compressed files, incorrect extensions files and etc., that are within time frame-critical search; which usually do not show up in standard searching methods. This function will not only enhance the searching methodology, instead it will make any tool to be more prominent than others of same type in the market.

3.5.7. Password Recovery

Tools considered here do not have password recovery mechanism for Windows and other platform only for Palm OS devices (Device Seizure version 1.0).

Conclusions

With the increasing use of smart phones for daily communication, the importance of digital evidence is increasing as well. Usually information systems i.e. internal memory or external memory and SIM card, leave many tracks, which can serve as evidence in a fight against crimes. Although digital data is relatively easy to find with few click (with specialized forensic tool) but they are also very sensitive e.g., the date of last entry into a file or text in e-mail messages is easily modifiable and file extensions can be forgeable. Thus, sensitivity of digital tracks plays a role in collecting and analyzing the digital evidence. Obviously, the evidence should not change and it is important that no evidence is lost. Therefore, tool developer need to have their hardware forensically certified or globally standardized to avoid any mishaps.

The reliability and verifiability of the functioning of a forensic software tool is of great importance because using a software tool is usually part of a more elaborate forensic investigation. Thus obtaining the results in timely manners is very crucial to some cases, which is why it is challenging for tool developer to keep their tool-supports up-to-date for any upcoming new smartphone(s) technology e.g. MSIM, SupperSIM support.

References

3GPP (1999). 'Technical Specification Group Terminals Specification of the Subscriber Identity Module -Mobile Equipment (SIM - ME) interface'. Available from: http://www.3gpp.org

Ayers, R. & Jansen, W. (2006). 'Forensic Software Tools for Cell Phone Subscriber Identity Modules'. National Institute of Standards and Technology

Ayers, R. & Jansen, W. (2004). 'Guidelines on PDA Forensics'. National Institute of Standards and Technology (NIST Special Publication 800-72)

Ayers, R. & Jansen, W. (2004). 'PDA Forensic Tools: An Overview and Analysis'. National Institute of Standards and Technology (NISTIR 7100)

Blue Mug [online] Available from: http://www.bluemug.com (25th June 2006)

Chevalier, S. Dang, H. Grance, T. & Kent, K. (2005). 'Guide to Computer and Network Data Analysis: Applying Forensic Techniques to Incident Response' (NIST Special Publication 800-86)

Constantinou, A. (2006). 'High capacity SIMs'. Informa Telecoms & Media

Electronics, 2006, How SmartPhone works? [online] Available from: http://electronics.howstuffworks.com/smartphone.htm (1st September 2006)

ETSI, 2006, "European Telecommunications Standards Institute", http://www.etsi.org/, cited 8 April 2006.

Focus Group, 2006, http://www.fft.uk.com/telecom_investigations.htm, cited 5 March 2006

Microsoft Mobile Solutions (2003). 'Deploying Microsoft mobile solutions – An overview'. Microsoft Corporation

Mislan, R. Small Scale Digital Device Forensics [online] Available from: http://www2.tech.purdue.edu/cpt/courses/CPT499D (18 September 2006)

mSystem. mSIM high-density SIM card [online] Available from: http://www.m-systems.com/site/en-US/Products/MegaSIM/MegaSIM (18 June 2006)

NCTP, 2006, National Cybercrime Training Partnership. [online] Available from: http://www.nctp.org (8th September2006)

Paraben Handheld Digital Forensics tools [online] Available from: http://www.paraben-forensics.com/handheld_forensics.html (1st September 2006)

Radio Tactics [online] Available from: http://www.radio-tactics.com/forensic_sim_detail.htm (1st September 2006)

Subscriber Identity Module [online] Available from: http://en.wikipedia.org/wiki/Subscriber_Identity_Module (1st September 2006)

RUIM, 2006, What is a RUIM Card? [online] Available from: http://www.wisegeek.com/what-is-a-ruim-card.htm (1st September 2006)

[49] Willassen, S. (2005). 'Forensic Analysis of Mobile Phone Internal Memory'. IFIP International Conference on Digital Forensics, National Center for Forensic Science, Orlando, Florida, USA

Oscar – Using Byte Pairs to Find File Type and Camera Make of Data Fragments

Martin Karresand[1,2], Nahid Shahmehri[1]

[1] Dept. of Computer and Information Science
Linköpings universitet,
Linköping, Sweden
[2] Dept. of Systems Development and IT-security
Swedish Defence Research Agency,
Linköping, Sweden
{g-makar,nahsh}@ida.liu.se

Abstract Mapping out the contents of fragmented storage media is hard if the file system has been corrupted, especially as the current forensic tools rely on meta information to do their job. If it was possible to find all fragments belonging to a certain file type, it would also be possible to recover a lost file. Such a tool could for example be used in the hunt for child pornography. The Oscar method identifies the file type of data fragments based solely on statistics calculated from their structure. The method does not need any meta data to work. We have previously used the byte frequency distribution and the rate of change between consecutive bytes as basis for the statistics, as well as calculating the 2-gram frequency distribution to create a model of different file types. This paper present a variant of the 2-gram method, in that it uses a dynamic smoothing factor. In this way we take the amount of data used to create the centroid into consideration. A previous experiment on file type identification is extended with .mp3 files reaching a detection rate of 76% with a false positives rate of 0.4%. We also use the method to identify the camera make used to capture a .jpg picture from a fragment of the picture. The result shows that we can clearly separate a picture fragment coming from a Fuji or Olympus cameras from a fragment of a picture of the other camera makes used in our test.

Keywords: Camera recognition, computer forensics, data recovery, file type identification, 2-gram frequency distribution.

1 Introduction

Being able to survey the contents of a hard disk is vital in, for example, a forensic examination or during recovery of lost data. Unfortunately fragmentation together with a corrupt file system complicates the mapping out of the hard disk. Depending on the level of fragmentation the task can be anything from hard to almost impossible. To further complicate the situation, there are, to the

best of our knowledge, no tools [1,2,3,4,5,6,7] available that can identify the file type of a data fragment without relying on meta data in some form. Due to this lack of adequate tools a forensic examiner can miss fragments of .jpg pictures containing child pornography on a hard disk or other storage media.

If all fragments of a certain file type can be found, the fragments can be used to recreate the original files. A method [8] to do the latter exists, but this work aims to provide a solution to the first part.

To be able to determine the file type of a data fragment without using meta data we have to look at the structure and contents of the fragment. For text-based data this is trivial, but for binary data we cannot rely on finding readable strings to use. The structure of the data itself is, however, unique for different types of files. This is true even for compressed files, which can be seen in Fig. 1 showing histograms of the byte frequency distribution of typical .jpg and .zip files.

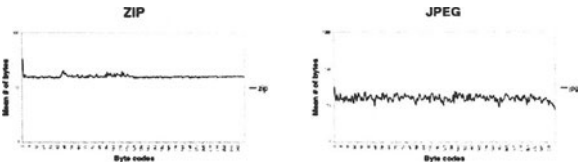

Figure 1. Histograms calculated over a number of 4 kB blocks of .jpg and .zip files. The headers of the .jpg files were stripped of before calculating the statistics. A logarithmic scale is used for the Y-axis.

The Oscar method [9,10] has been proposed as a method to identify the file type of arbitrary data fragments in a way that is fast enough to be used as the main tool for forensic examinations of current and future storage media. The first version of the method uses 1-grams, i.e. single bytes, to create a histogram of the byte frequency distribution (BFD) of a data block, and to measure the rate of change (RoC) of consecutive bytes. The Oscar method has been updated [11] to use byte pairs, 2-grams, instead of single bytes. The first test results indicated that the method made it possible to tell some digital camera makes apart using 4 kB picture data fragments.

This paper deepens the camera recognition tests, first presenting the necessary background. The implementation of the 2-gram method has been slightly updated and we perform the tests using larger amounts of data for each camera make.

2 Method

The Oscar method is built measuring the distance between an unknown sample and a model (called centroid) of a specific file type. The centroid contains the

mean frequency and the corresponding standard deviation values of the bytes or byte pairs of the file type.

2.1 2-gram Oscar

The centroid modelling a file type is created by counting the number of unique 2-grams in a large number of 1 MB blocks of data of a specific file type. The data in each 1 MB block is used to form a frequency distribution of all 65536 different possible 2-grams, and the mean and standard deviation of each 2-gram is calculated and stored in two $256 \star 256$ matrices. The values are then scaled down to correspond to a block size of 4 kB.

The 4 kB size is used because it is equal to the common size of memory pages and disc clusters in current home computers. The reason for using 1 MB data blocks is to get a solid foundation for the mean and standard deviation calculations. When using a block size less than the possible number of unique 2-grams the calculations inevitably become unstable.

When the similarity between a sample fragment and a centroid is measured the same type of quadratic distance metric as for the 1-gram Oscar method is used. Equation 1 describes the metric, where matrix S depicts the sample 2-gram frequency distribution and C the mean value matrix of the centroid. The standard deviation of 2-gram ij is represented by σ_{ij}. There is also a smoothing factor $\alpha = 0.000001$, which is used when $\sigma_{ij} = 0$ to avoid division by zero.

$$\mathrm{d}(S,C) = \sum_{i=0,j=0} (s_{ij} - c_{ij})^2 / (\sigma_{ij} + \alpha) . \tag{1}$$

For the camera recognition experiments the smoothing factor α is calculated as

$$\alpha = \frac{1}{2^n} \qquad n = [1, 2, \ldots, 60], \tag{2}$$

where n is the size of the data in MB. For sizes larger than 60 MB $n = 60$.

The quadratic term is used to favour smaller deviations over larger. As can be seen in Eq. 1 the differences are weighted by the standard deviation of each 2-gram. This is done to lessen the influence of 2-grams that vary significantly and in that way only use the key features of the 2-gram distribution.

The distance value of a sample is compared to a empirically predetermined threshold. Samples giving distance values below the threshold are categorised as being of the same file type as the centroid.

The reason for not using a more advanced distance metric is execution speed. The current maximum size of a hard disk is 750 GB and hence a forensic examination can involve a tremendous amount of data to scan.

2.2 Advantages and Disadvantages

There are both advantages and disadvantages to using 2-grams. The disadvantages of using 2-grams are the exponential increase of the centroid's size and the

lower precision in measuring distance, due to the scaling down of the values of the centroid.

The size of the centroid mainly affects the execution speed of the algorithm, but also the memory footprint. In theory the effect is a 256-fold increase in processing time and memory footprint compared to the original 1-gram methods. The increase in memory footprint can be ignored due to the amount of RAM used in modern computers; the requirement of the current implementation is a few hundred kB of RAM. By optimising the algorithm the processing time can be decreased, but the decrease is not bounded.

The problem of scaling down the values cannot be disregarded. If we assume a uniform distribution of byte pairs in a file and look at a 4 kB fragment of that file, the probability of a specific byte pair is $\frac{4096}{65536} = \frac{1}{16}$. Each time a byte pair is found in the 4 kB fragment, it equals finding 16 such byte pairs in a 64 kB fragment. This renders the distance measurement inaccurate and increases the risk of detection errors. By using fragments of 1 MB (16 times larger than 64 kB) when creating the centroid we lessen the impact of the down-scaling.

The main advantage of the 2-gram method is the automatic inclusion of the order of the bytes into the method, signalling that an unknown block does not conform to the specification of a specific file type and thus should not be categorised as such. A typical example is the appearance of certain .jpg header byte pairs disallowed in the data part. Another example is special byte pairs never occurring in files created by a specific camera or software.

3 Evaluation

The experiments presented in [11] have been extended with one more camera make for the camera recognition and a new file type for the file type identification. The experiments are described in the following subsections.

3.1 File type identification

The files for the experiment were taken from an office computer running Windows XP, fully patched. The individual files were padded with zeros to adjust their lengths to a multiple of 4 kB.

The .jpg file centroid, zip file centroid, and Windows executable centroid used in earlier experiments were complemented by an .mp3 file centroid. The detection rate of each centroid was then measured together with the corresponding false positives rate.

The .jpg centroid was created from 104 MB of picture data, the headers were stripped of the original pictures before we created the centroid. For the .mp3 centroid we used 859 MB of data. The zip file centroid was made from 104 MB of different zip-based files, created using WinZip and the Linux gzip and zip utilities. The complete zip files were used. The Windows executable file centroid was created from a 147 MB large concatenation of different files, all being categorized as either "MS-DOS executable (EXE), OS/2 or MS Windows",

or "MS-DOS executable (EXE)" by the Linux *file-4.12* utility. All such files begin with the magic number 0x4D5A, "MZ".

The experiment was performed using a 72 MB test file consisting of 57 files of 51 different types concatenated into one. The templates used as ground truth for the experiment labelled the .jpg headers as non-.jpg. The executable, mp3 and zip files were labelled as true positives in their entirety. The distribution of 4 kB fragments in the test file was 1150 .jpg, 4397 Windows executables, 1426 .zip, 1235 .mp3, and 10275 fragments of other types.

3.2 Camera recognition

The camera recognition experiment was performed using an 80/20 model, i.e. we created a centroid from 80 % of the picture data for a specific camera and used the remaining 20 % for testing. Consequently, there were five sub-experiments, which were used to calculate a mean value for the detection rate. Each centroid was tested against all different cameras giving us the Cartesian product of the picture and centroid combinations.

The amount of data used to create the centroids and to test them were larger than in the previous camera recognition experiment, where the centroids were created from 8 MB of data. This time we used all data available, ranging from 14 MB to 1.5 GB, with a mean of 335 MB.

The different cameras used for the test are shown in Table 1.

Table 1. The camera make and models used for the experiments

Canon DIGITAL IXUS 400	Canon DIGITAL IXUS v
Canon PowerShot A70	Casio EX-Z40K
Fuji FinePix2400Zoom	Fuji FinePix E550
Kodak CLAS Digital Film Scanner/HR200	Konica KD-310Z
Konica Minolta DiMAGE G400	Konica Minolta DiMAGE X60
Nikon D50	Nikon E3500
Olympus D600	Sony DSC-P8

The pictures are all standard family photographs, both indoor and outdoor, with varying lighting conditions and settings. Some of the pictures are rotated using the camera's built-in editing facility, otherwise the pictures are unedited.

4 Result and Discussion

The .jpg and .mp3 type files both had detection rates above 75%, with less than 0.5% false positives. Regarding the camera recognition experiment, the Fuji and Olympus cameras stand out clearly. All other cameras are indistinguishable, which were not the case in the previous camera recognition experiment.

4.1 File type identification

The results of the file type recognition experiment can be seen in Fig. 2.

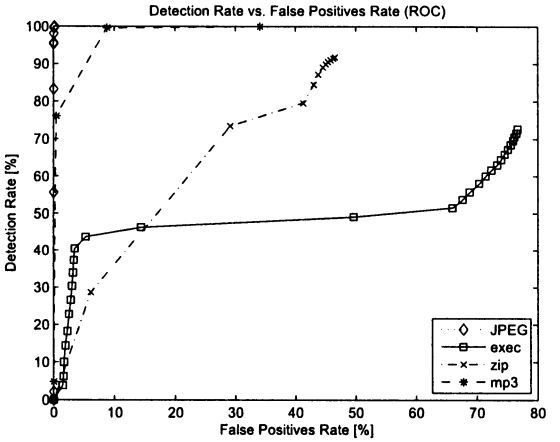

Figure 2. The detection rate vs. the false positive rate using .jpg, Windows executable, mp3 and zip centroids

The 2-gram method reaches 100% detection rate for .jpg file fragments with a false positive rate of 0.1% and there are no false positives until the detection rate reaches 98.1%. The .mp3 curve is only slightly lower than the .jpg, giving a detection rate of 76% with a false positives rate of 0.4%

Since the false positive rate of the zip files and the Windows executable files are almost or higher than 50% these cannot be used as they are. We do, however, think that the results can be improved by fitting the selection of data for centroid creation to the standards connected to the file types.

4.2 Camera recognition

Fragments of pictures taken by the Fuji and Olympus cameras are clearly distinguishable from picture fragments of the other cameras. The graphs of the two groups of cameras are almost identical within the groups. Hence the hypothesis presented in the previous paper [11] on the Oscar method, saying that it would be possible to tell cameras apart by looking at the structure of their pictures, does not hold.

Figure 3 shows the signature matrices coded as bitmaps for the different camera makes. The colour saturation of the bars represents the number of true positives: the darker the bar, the higher the level. The centroids, i.e. bars, correspond to, from left to right (see also Table 1): Casio, Konica G400, Nikon

E3500, Sony, Canon IXUS 400, Fuji 2400, Nikon D50, Canon A70, Olympus, Konica 310, Konica X60, Fuji E550, the scanner, and Canon IXUS v.

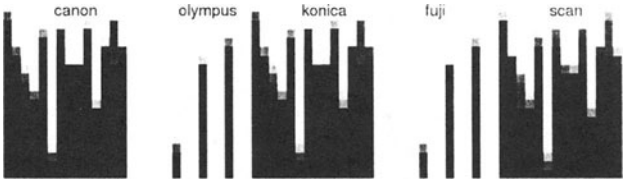

Figure 3. A comparison chart for pictures mapped by different centroids. Since the results can be grouped into two groups we only show five graphs.

As can be seen in Fig. 3 the Casio camera centroid has a better detection rate than the other centroids. The scanner centroid gives the highest detection rate for its own pictures, which might be an indication of a possibility to detect scanned picture fragments. The Fuji and Olympus pictures contain a special marker sequence that the other pictures do not, consequently only these centroids can detect them. The Canon A70 and Konica 310 graphs in the figure are used to represent the other cameras' graphs, which all look more or less the same.

4.3 Discussion

The detection rate and false positives rate of the Windows executables centroid are poor to say the least, in fact, they are even worse than guessing. There are, however, a large number of different file formats fitting our requirement that the files should start with the "MZ" byte pair. The structure also varies within the files themselves. Typically there can be sections of readable ASCII characters, well structured binary tables, and sections of machine code, arbitrarily ordered.

Due to the limited set of cameras used for the experiments we cannot really say anything more than that it is possible to distinguish Fuji and Olympus picture fragments from picture fragments of the other cameras included in the experiment. The reason for Fuji and Olympus being recognizable are their use of restart markers in the data, which make camera recognition trivial. Since also some scanner software add restart markers to their files, the group will grow larger and consequently the camera recognition ability will diminish.

The large amount of data in this experiment, in combination with the dynamic α value (see Eq. 2), are the main reasons for the uniform result. What can be discussed is whether the stabilising feature of the α is preferable or not. If we compare this experiment to the previous camera recognition experiment, we still see some minuscule differences in the detection rates. Are these differences related to the make of a camera and if so, can they be robustly enhanced and detected? To what extent do the calculation of α suppress these differences?

5 Related Work

The works presented in this section are related to the 2-gram method either by the use of an n-gram algorithm for operation on fragmented data in different ways, or by their ability to identify the camera used to capture a picture without relying on make and model information in the file header.

The 1-gram Oscar method [9,10] uses two algorithms, byte frequency distribution and rate of change, to identify the file type of a data fragment on byte level. For BFD the mean and standard deviation of each byte frequency is used to form a model, called centroid, of a specific file type. The similarity of an unknown sample and the centroid is measured by calculating the difference between each byte value frequency of the sample and the centroid, using a sum of squares weighted by the standard deviation of each byte frequency.

The rate of change (RoC) algorithm measures the absolute value of the difference between consecutive bytes. This approach allows the smoothness of the byte stream of a data block to be taken into consideration, decreasing the number of false positives from data blocks that are of a different file type from the centroid, but which have a similar byte frequency histogram.

The 2-gram Oscar method [11] differs from the 1-gram method by the fact that the latter does not automatically and fully take special byte combinations (required or disallowed) into consideration. Nor does it consider if the rate of change between bytes is positive or negative.

An n-gram based method that is related to the 2-gram method is the intrusion detection system PAYL [12,13] from the research group lead by professor Stolfo at Columbia University, USA. The same group has also used n-grams for file type recognition using fileprints [14,15]. The main differences between these methods and the 2-gram method are the previous methods' dependency on file header information, as well as the fact that Stolfo and his coauthors do not consider byte order, which the 2-gram method does.

Also McDaniel and Heydari [16] use the concept of n-grams in a file type identification method. They use different statistical measures applied to single bytes in files to make finger prints of files. McDaniel and Heydari's main source of data is the header and footer parts of files. They try to consider the ordering of the bytes in files through the use of what they call *byte frequency cross-correlation.* One example of bytes with high byte frequency cross-correlation is the html tag markers "<" and ">". The McDaniel and Heydari method differs from the 2-gram method through the use of single bytes, as well as the their method's dependence on file header and footer data.

Yet another n-gram based method is presented in a paper by Shanmugasundaram and Memon [8]. They use a sliding window algorithm for fragmented file reassembly using n-gram frequency statistics. They let the window pass over the edges of the fragments and then combine those fragments having the highest probability of being consecutive based on the frequency of the n-grams found in the window. Their use of n-grams frequency statistics is similar to the 2-gram method, but they apply the method in another way and require all fragments of a file to be known in advance. Therefore the 2-gram method is applicable to

the steps before Shanmugasundaram and Memon's method and is only partly related to their method. The two methods can, however, be joined together and used to first identify and then reconnect fragments of files.

Camera recognition applied to the pixel level of pictures is presented by Lukáš, Fridrich, and Goljan [17,18,19]. They base their method on the small deviations in uniformity in the CCD of a camera. The deviations are detectable even after compression and other picture manipulation exercises. In the papers they show that the deviations are unique to a specific camera. Another camera recognition method applied to the pixel level of pictures is presented by Kharrazi, Sencar and Memon [20]. They identify 34 features that can be used to identify which camera make and model was used to capture a picture. These two methods differ from the 2-gram method and its camera recognition ability in that both methods are applied to unfragmented, complete pictures and the 2-gram method works on fragmented data.

6 Conclusion and Future Work

In this paper we used an variant of the 2-gram Oscar method to identify the camera make used to capture a picture. Our preliminary results show that Fuji cameras as well as Olympus differ significantly from other camera makes and thus can be identified from a picture fragment. This is however directly related to the use of restart markers in the data, making the recognition trivial. We also tested the 2-gram method on .mp3 data fragments giving a detection rate of 76% with 0.4% false positives.

There is still work to be done in developing centroids covering more file types. We also need to further investigate the reason for the poor results of the executable file centroid. One reason can be that our definition of executable files is too general, since the current definition covers a large number of sub-types. We also need to experiment with executables compiled for platforms other than Windows XP on IA32.

As a natural continuation of the work with the Oscar method we will look into the problem of recreating parts or complete original files from the fragments found.

References

1. CONVAR Deutschland: Pc inspector. `http://www.pcinspector.de/file_recovery/uk/welcome.htm` accessed 2005-10-31.
2. Carrier, B.: The Sleuth Kit. `http://www.sleuthkit.org/sleuthkit/index.php` accessed 2005-10-25.
3. Farmer, D., Venema, W.: The Coroner's Toolkit (TCT). `http://www.porcupine.org/forensics/tct.html` accessed 2005-10-25.
4. Guidance Software: Encase forensic. `http://www.guidancesoftware.com/products/ef_index.asp` accessed 2005-10-31.
5. QueTek Consulting Corporation: File scavenger. `http://www.quetek.com/prod02.htm` accessed 2005-10-31.

6. iolo technologies: Search and recover. http://www.iolo.com/sr/3/ accessed 2005-10-31.
7. ilook-forensics.org: ILook Investigator Forensic Software. http://www.ilook-forensics.org/ (2005) accessed 2006-04-28.
8. Shanmugasundaram, K., Memon, N.: Automatic reassembly of document fragments via context based statistical models. In: Proceedings of the 19th Annual Computer Security Applications Conference. (2003) http://www.acsac.org/2003/papers/97.pdf, accessed at 2006-04-30.
9. Karresand, M., Shahmehri, N.: Oscar – file type identification of binary data in disk clusters and ram pages. In: Proceedings of IFIP International Information Security Conference: Security and Privacy in Dynamic Environments (SEC2006). Lecture Notes in Computer Science (2006) 413–424
10. Karresand, M., Shahmehri, N.: File type identification of data fragments by their binary structure. In: Proceedings from the Seventh Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop, 2006, Piscataway, NJ, USA, IEEE (2006) 140–147
11. Karresand, M., Shahmehri, N.: Oscar – file type and camera identification using the structure of binary data fragments. In Haggerty, J., Merabti, M., eds.: Proceedings of the 1st Conference on Advances in Computer Security and Forensics, ACSF, Liverpool, UK, The School of Computing and Mathematical Sciences, John Moores University (July 2006) 11–20
12. Wang, K., Stolfo, S.: Anomalous payload-based network intrusion detection. In E. Jonsson el al., ed.: Recent Advances in Intrusion Detection 2004. Volume 3224 of Lecture Notes in Computer Science., Springer-Verlag (July 2004) 203–222
13. Wang, K., Cretu, G., Stolfo, S.: Anomalous payload-based worm detection and signature generation. In Valdes, A., Zamboni, D., eds.: 8th International Symposium on Recent Advances in Intrusion Detection, RAID 2005. Volume 3858 of Lecture Notes in Computer Science., Springer-Verlag (2006) 227–246
14. Stolfo, S., Wang, K., Li, W.J.: Fileprint analysis for malware detection. Technical report, Computer Science Department, Columbia University, New York, NY, USA (2005) Review draft.
15. Li, W.J., Wang, K., Stolfo, S., Herzog, B.: Fileprints: Identifying file types by n-gram analysis. In: Proceedings from the sixth IEEE Sytems, Man and Cybernetics Information Assurance Workshop. (June 2005) 64–71
16. McDaniel, M., Heydari, M.: Content based file type detection algorithms. In: HICSS '03: Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 9, Washington, DC, USA, IEEE Computer Society (2003) 332.1
17. Lukáš, J., Fridrich, J., Goljan, M.: Determining digital image origin using sensor imperfections. In: Proceedings of SPIE Electronic Imaging, Image and Video Communication and Processing. (2005) 249–260
18. Lukáš, J., Fridrich, J., Goljan, M.: Digital bullet scratches for images. In: Proceedings of ICIP 2005. (2005)
19. Lukáš, J., Fridrich, J., Goljan, M.: Digital camera identification from sensor pattern noise. IEEE Transactions on Information Forensics and Security **1**(Summer 2) (2006) 205–214
20. Kharrazi, M., Sencar, H., Memon, N.: Blind source camera identification. In: Proceedings of ICIP '04. 2004 International Conference on Image Processing. Volume 1. (2004) 709–712

An empirical methodology derived from the analysis of information remaining on second hand hard disks.

Grigorios Fragkos[1], Vivienne Mee[1], Konstantinos Xynos[1], Olga Angelopoulou[1]

[1]Information Security Research Group
University of Glamorgan,
CF37 1DL, Wales, UK
{gfragkos, vmmee, kxynos, oangelop}@glam.ac.uk

Abstract. In this paper we present the findings of an analysis of approximately 260 second hand disks that was conducted in 2006. A third party organisation bought the disks from the second hand market providing a degree of anonymity. This paper will demonstrate the quantitative outcomes of the analysis and the overall experiences. It will look at how analysts can expand their tools and techniques in order to achieve faster results, how one can organise the analysis based on the way information is found and finally a holistic picture of the case should be generated following the proposed methodology.

Keywords: disk analysis, second hand disks, corporate data, computer forensic analysis, Forensic Race

1 Introduction

During 2006 we organised a research initiative on the analysis of residual data on second hand hard disks. In this respect a number of hard disks where studied in order to identify and understand the data that can be revealed from randomly purchased hard disks from a number of different countries. The sample countries/regions included the U.K., Germany and North America.

It is vital to be aware of the importance and utilisation of a disk study for the computer forensics science [1]. The researcher gains awareness concerning end-user and corporate knowledge regarding data exposure. Unaware users can become victims of espionage and blackmail. By analysing disks that contain any personal data for research reasons the possibility of espionage is prevented, while on the other hand people that hear about this study get informed about the way their data can be retrieved and used for malicious reasons. Therefore, from the researcher's point of view, potential criminal activity can be identified where sensitive data is available and can be used for such purposes.

The overall number of disks analysed lead to results that should be considered further, such as statistical reports concerning the users' familiarity with techniques like wiping data from their hard disks and thus securing their privacy when disposing of disks. In any case that data is left behind in hard disks there is a fraud risk for the user. Consequently, the capacity of personal information revealed can even lead to Identity Theft by thoroughly profiling the victim, as there were multiple cases where personal identification details could be retrieved.

This disk study signifies issues not only for the research community, but for the general user awareness as well. Further to discussing those issues based on our experience of taking part in this disk study, in this paper we propose a methodology that could be implemented in order to simplify and manage a disk study research procedure.

2 Forensic Race

As a result of the disk study, a new terminology, "Forensic Race" was brought about. The word race differentiates the species from one kind to the other. Consequently, the term "Forensic Race" determines the race of a computer's hard disk or any type of digital data repository. The "Forensic Race" of a hard disk is based on a hard disk's capacity, operating system and in combination with the nature of the data contained within the disk drive.

In the disk study, it was found that the majority of the disks were between 500 Mega Bytes and 10 Giga Bytes with an overwhelming 69 per cent (see Fig 1). From the disk's capacity it is possible to assume approximately the year from which it was manufactured. This, most likely can then lead the analyst to a simple deduction that the Operating System would be from around that period. This was validated since the majority of the systems contained old Operating Systems (e.g., MS Dos, MS. Windows 98, NT, 2000 and ME).

As a result of the 'Forensic Race' term, an investigator could predict the approximate time period required. During an analysis, the size of the disks and the amount of information stored on them contributes to this variable. The majority of the disks had low capacity therefore the time period took for each one to be investigated was substantial. This is not always holds true since the demand to store more and larger files requires larger hard disks. Consequently, future disk studies will consist of disks that will have massive capacity. Therefore, the investigation will demand a time management schedule in order to contact the analysis in a reasonable period of time.

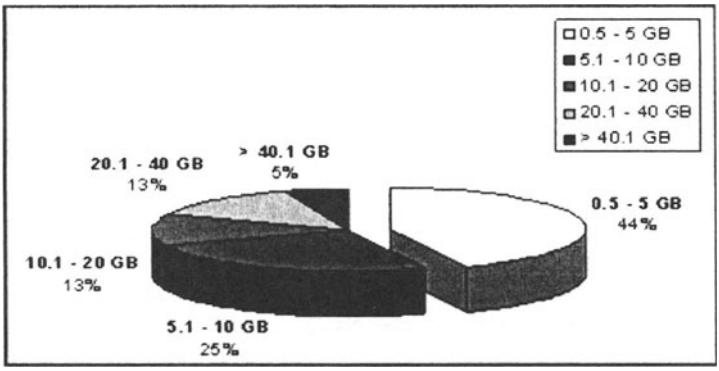

Fig. 1. Disk Capacity Percentages

With the time factor in mind, it is very helpful, from an investigators point of view, to have an idea of the race of the disks you are dealing with in advance. Thus, it is makes sense to know is you are dealing with a Dwarf or a Gnome or if you are dealing with a Cyclops or a Giant (see Table 1). What one wouldn't want to deal with is a Black Giant.

Table 1. Classification sample of a disk's Forensic Race

Giant	Large Capacity – Vast amount of data present
Cyclops	Large Capacity – Very few data present
Dwarf	Low Capacity – Vast amount of data present
Gnome	Low Capacity – Very few data present
Blue	Corporate data present
Green	Individual data present
Black	Illicit Material

A hard disk can be classed as a Cyclops, as a Dwarf, as Gnome etc.

- The *Cyclops* is a hard disk that has a large capacity that even a one eye person could easily see that it contains almost no data and definitely no illicit material.
- On the contrary, a *Dwarf* is a small capacity hard disk that contains a vast amount of data that would take days to investigate. The disks classify themselves according the data they do contain.
- A *Gnome* is a small capacity hard disk that contains a small amount of legitimate data.

It is possible to expand the list of the Forensic Race categories depending on how detailed the investigator would like to categorise findings of an investigation.

3 A proposal for an empirical approach

The methodology of computer forensics investigation calls out for a more empirical approach. Instead of just following an out of the book procedure, while investigating a case, the consideration of past experience along with an empirical approach could lead to a far more efficient analysis. The required experience and knowledge to deal efficiently with the vast amount of information contained within the disks, has been developed through the investigation of a series of data repositories in the past. However, the time allocated for the analysis was 3 months for approximately 260 disks. Within that time the disks had to be imaged, recovered, data carved, cross referenced with existing data or data already recovered from other disks, characterise and categorise the nature of the data recovered.

After completing the disk study [2] [1] the need for a more flexible methodology was raised. The investigators felt that they shouldn't arrange the disks to be analysed based on their criteria but let the disks decide which one it is going to be investigated in what order.

The Table 2 represents the total number of disks provided and how this number went down by excluding disks that matched certain criteria.

Table 2. Disks obtained / analysed

Disk Category		Excluding List	
Total Number of Disks	**259**	-124	*Faulty / Unreadable*
Readable Disks	**135**	-70	*Wiped*
Disks Containing Data	**65**	-4	*Handed to the police*
Final Disk Investigated	**61**		

Comparing the total numbers of disk provided (259) with the number of the disks that has been actual investigated (61) it is clearly that the only the $^1/_4$ (23.5%) of the disks were processed (see Table 2).

The proposed methodology tries to generate an informative table that will provide with some internal information of how a disk study should start and in which order to be conducted.

1. Start by excluding the faulty disks, this is the easiest part. Indisputably, these are the one that cannot be imaged due to physical damage and need special equipment or resources in order to be investigated.
2. An automated procedure that checks for wiped drives that contain no data is the second step of the methodology.
3. The third step is more complex and advanced than the previous steps. It involves four different applications which will sift through the date on the disk images.
    - The first application will try to locate (within the present data) and/or recover (deleted) any pictures (photographs, image files) of

any type. By examining the different headers and footers of various image files, the application can extract full or partial known image file types. Researchers form the Information Research Group (ISRG) within the University of Glamorgan conceived the idea and developed a program which proves that this is possible.

- The second application will extract all the thumb.db files by locating them or recovering them from the slack space. Then a smart application goes through the thumb.db files and extracts all the pictures stored in them. The special case with the thumb.db file is that it stores forever (until wiped) a backup preview version of the user's pictures even if they have been deleted or even wiped. Furthermore, a small preview of a picture is more likely to be found when stored with others in one single file, than trying to recover the large version of a picture spread across the hard disk and most possibly been overwritten in various parts. Researchers from the Information Security Research Group (ISRG) within the University of Glamorgan have also conceived this idea and have developed a program [3] which proves that this is possible. Recently, a tool called Vinetto [7] was released that extracts pictures and generates reports too.
- The third application extracts the directory names form the index.dat file(s) (see Fig. 2) and tries to locate/recover the contents. This is where the Operating System (OS) is caching the user's activity. The conceived idea that extracts the thumb.db files was slightly modified to extract the contents (directory names and URLs) of the index.dat files.
- Last but not least, the fourth application that tries to extract the registry file of the user. Further investigation in the registry can reveal a vast amount of information if needed that could totally end up profiling the user. Information such as software installed, hardware installed, user's Internet activity and habits can all be retrieved from the Registry alone [4].

Once these four steps have been completed, the informative table of internal information gives the investigator a brief overview of what the investigation is going to entail, and which disks have priority over others for further analysis.

3.1 Further details about the methodology

By having a collection of the pictures contained into the disk it is fairly easy to identify potential illicit material within the hard disk. Thus, in a very small period of time the pictures will do their best to help the investigators identify disks that has to be reported to the police immediately.

Especially with the MS Windows® OS, a list of the original files that are shipped with the Operating System can be produced. For every MS Windows® distribution a

list of MD5 hashes of each file can be generated. Thus, during the investigating process, a number of approximately 50,000 files (depends the distribution) can be excluded from the processing cycles of the investigation's procedure. Additionally, this will also minimise the successful hits returned when searching keywords by focusing at the unknown for the system files which however are the user's data.

The registry file contains many secrets that are waiting to be revealed. A quick search through specific locations in the registry, if this is possible of course, could reveal the origins of the disk. The company name used to register software, the computer name, the type of installed programs, quick list of URLs visited; network shares and previously connected devices provide a good chance to the investigator to identify if the data present belong to some kind of company, to some individual as home user or both (see Table 3).

Table 3. Origin of disks investigated

Disks Investigated	Commercial Data Present	Individual data present
65 (Illicit material)	37	28

4 Advantages of the proposed methodology

Taking under consideration all this information revealed by the automated process and by having in mind the idea of how to classify the "Forensic Race" of a disk. It is fairly possible to start organising the investigation according to the estimated time that has to be spent on each case. The approximate idea of which disk contains what kind data can easily lead us to start with the larger in capacity disks that seems to contain a lot of corporate data and continue with the smaller in capacity disks that also seems to contain data from companies. That way we can leave all the disks that seem to contain data about individuals for the end of the investigation. Additionally, large disk that have been identified to contain very few data can be also left for the end of the investigation. Finally, disks that seem to have been originated from the same source (i.e. same company) can be grouped together from the beginning of the investigation despite their capacity.

By checking the time spend for each disk to be investigated in a random order to the overall time spend to complete the disk study one could see that the overall timeframe could be minimised. There have been disks that were only 6 GB in capacity and took ten times more time to investigate than disks that where double in size. Having an approximate idea of how long could take you to investigate a disk can be really helpful while organising such large scale projects.

- Disks that were found to belong to a specific group (i.e. from the same individual or the same company) should be crossed referenced every time new data were identified. Another example would be disks that have foreign origin.

- It is also very convenient to know from the beginning that you have to deal with a number of disks in different foreign languages. Consequently, these disks will be allocated to the proper people in order to perform the investigation in that foreign language. Otherwise in the middle of a study could end up trying to find a translator.

4.1 Providing some examples

The process of extracting information from the thumb.db files and index.dat files is similar.

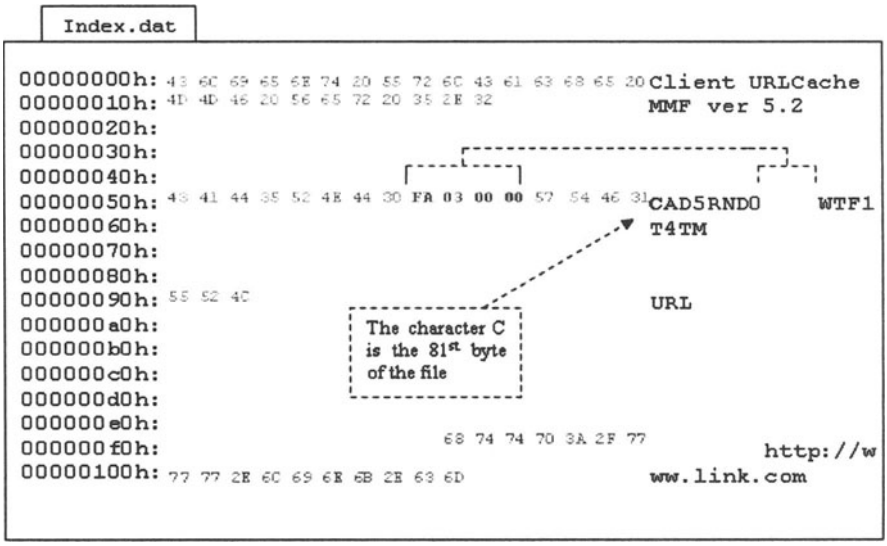

Fig. 2. Overwiew of the Index.dat file

Within the Fig. 2 it is clear how the information is stored in the index.dat file. A pattern is used that helps the investigator to extract information. After 80 bytes form the begging of the file the name of the directories starts. Every directory consists of 8 capital alphanumeric characters. Between the directory names the hex values FA 03 00 00 separates them with each other. CAD5RND0 and WTF1T4TM are the random generated directory names. Furthermore, going through the file searching for the URL string a web link will be found after 100 bytes (see Fig 3).

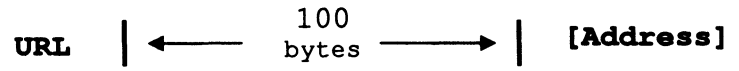

Fig. 3. *Pattern of extracting the web addresses*

There are numerous locations by which information can be extracted to give an analyst a further understanding of what data is present on the disk. As mentioned before it is possible to get usernames, company names and a user's internet activity all

just solely from the Registry. Mee et al [5] have given some examples of what and where this information can be extracted from the Registry. The following tables summarises these examples illustrating the keys which hold the information.

Table 4. Registry Six main Root Keys with their associated data [7]

Root Key	Description
HKEY_CURRENT_USER	Current logged-on user data
HKEY_USERS	Data about all the user accounts on the machine
HKEY_CLASSES_ROOT	File association and Object Linking and Embedding (OLE) registration information
HKEY_LOCAL_MACHINE	System related information
HKEY_DYN_DATA	Performance data
HKEY_CURRENT_CONFIG	Information about the current hardware profile

Table 5. System information which can be extracted

HKEY_LOCAL_MACHINE\Hardware\Description\System\CentralProcessor\0\ProcessorNameString	Name of the processor that the system in running
HKEY_LOCAL_MACHINE\Hardware\Description\System\ system	Bios that the machine is running the bios date of both the video bios and the system bios.

Table 6. Illustrates sample locations in the registry whereby Software related information can be extracted

HKEY_LOCAL_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion key	operating system installed, its service pack, system default path, and the registered owner can also be found in this Registry root key
HKEY_LOCAL_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion key \CSDVersion	service pack name that has been installed
HKEY_LOCAL_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion key \Winlogon	can be set to allow a user to automatically log onto the system, whenever the system boots

Table 7. Illustrates sample information related to specifically to users in which can be extracted from the Registry

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\Current Version\Profile List	lists all current and deleted users that has accesses to the machine, identified by their SID.
HKEY_USER*SID*\Software\Microsoft\Internet Explorer\TypedURLs	stores all the typed URLs that the user typed into Internet Explorer
HKEY_USER*SID*\Software\Microsoft\MSNMessenger\PerPassportSettings\	gives information about the location of the default path for received files for the user's MSN Messenger software

HKEY_USERS*sid*\Software\Microsoft\Messenger Service\ListCache\.Net Messenger Service	holds the entire user's MSN contact list
HKEY_USER*SID*\Software\Microsoft\SearchAssistant\ACMru\5683	user's recent searches in Windows

5 Conclusions

The proposed methodology of this paper is based on experience. Simple forensic applications could easily help the investigators to organise their study in advance when dealing with unknown quantities of data to be analysed. We believe that in the near future a system could be build that will be able to perform all the steps presented here automatically. Consequently, computer forensics investigators will be able to deal with a large number of high capacity disks in an efficient way.

5.1 Acknowledgements

Completing a disk study of approximately 260 disks is not a trivial task. Firstly, we would like to thank Dr. Andy Jones, Dr. Craig Valli, Dr. Iain Sutherland, Dr. Andrew Blyth, Paula Thomas for providing the disks and organising the disk study. In addition we would like to express our appreciation to our colleagues' researchers within the ISRG for the team effort to finish the disk study on time. Which include: Theodore Tryfonas, Michael Pilgermann, Stilianos Vidalis, Huw Read, and Abdulrazaq Al-Murjan. Last but not least, Vivienne Mee would like to acknowledge the support of EPSRC and DSTL.

References

1. Andy Jones, Craig Valli, Iain Sutherland, Paula Thomas (2006) An Analysis of Information Remaining on Disks offered for sale on the second hand market. Journal of Digital Security, Forensics & Law. Accepted for publication
2. Andy Jones, Vevienne Mee, Cristopher Meyler, Joanna Gooch, (2005), Analysis of Data Recovered from Computer Disks released for Resale by Organisations, Journal of Information Warfare
3. Fragkos G, Xynos K, (2004), "Thumb.db extractor – The research contacted and a proof of concept application that extract the pictures", [Online] http://www.s1m0n3.org/man0war/program.shtml
4. Mee, V, Tryfonas, T and Sutherland, I (2006), "The Windows Registry as a forensic artefact: Illustrating evidence collection for Internet usage", Digital Investigation, Vol. 3, Issue 4.
5. Mee, V., Jones, A., (2004), "The Windows Operating System Registry – A Central Repository of Evidence", Proceedings of the Information Warfare 2004 Conference
6. Mee, V., Jones, A., (2005), "The Windows Operating System Registry", Proceedings of the E-Crime and Computer Evidence
7. Roukine M., (2006), "Vinetto - a forensics tool to examine Thumbs.db files", [Online] http://vinetto.sourceforge.net/test_JF_Beckers/vinetto.html

Towards Trustable Digital Evidence with PKIDEV: PKI Based Digital Evidence Verification Model[i]

Yusuf Uzunay, Davut Incebacak, Kemal Bicakci

Informatics Institute, Middle East Technical University, Ankara, Turkey
{yuzunay, davut, bicakci}@ii.metu.edu.tr

Abstract. How to Capture and Preserve Digital Evidence Securely? For the investigation and prosecution of criminal activities that involve computers, digital evidence collected in the crime scene has a vital importance. On one side, it is a very challenging task for forensics professionals to collect them without any loss or damage. On the other, there is the second problem of providing the integrity and authenticity in order to achieve legal acceptance in a court of law. By conceiving digital evidence simply as one instance of digital data, it is evident that modern cryptography offers elegant solutions for this second problem. However, to our knowledge, there is not any previous work proposing a systematic model having a holistic view to address all the related security problems in this particular case of digital evidence verification. In this paper, we present PKIDEV (Public Key Infrastructure based Digital Evidence Verification model) as an integrated solution to provide security for the process of capturing and preserving digital evidence. PKIDEV employs, inter alia, cryptographic techniques like digital signatures and secure time-stamping as well as latest technologies such as GPS and EDGE. In our study, we also identify the problems public-key cryptography brings when it is applied to the verification of digital evidence.

1 Introduction

Today, criminals use information technologies intensively to facilitate their offenses, creating new challenges for the law enforcement agencies especially in effectively and securely searching for, capturing, and preserving all types of digital evidence potentially germane to an investigation. Digital evidence can reveal how a crime was committed, provide investigative leads, disprove or support witness statements, and identify likely suspects [1].

By definition, digital evidence stands for the information of value to a criminal case that is stored or transmitted in digital form [2]. Once most of the data that might be relevant to an investigation has been ferret out as evidence, the problem is how to prove that these evidences have not been altered since they were obtained. Digital data, by its fragile nature, can easily be disrupted, changed or replaced by affecting the legal admissibility of digital evidence. To maximize the evidential value of digital evidence in the time of trial, it is needed to establish some kind of assurance i.e.; to prove the integrity, nonrepudiation and authenticity of digital evidence.

In this paper, we propose a novel model called PKIDEV: PKI based Digital Evidence Verification model to provide the integrity and authenticity of the digital evidence captured from a crime scene. PKIDEV employs a lot of different authentication mechanisms ranging from digital signatures to biometrics and presents an integrated solution to overcome existing problems in verification of digital evidence.

This paper is organized as follows; in the subsequent section, we discuss the existing challenges with the digital evidence. Section 3 introduces some shortcomings of PKI and digital signatures. After briefly mentioning the previous studies in section 4, we explain our model PKIDEV in detail and describe how each component in the model interacts among themselves in section 5. In section 6, we present the benefits of the model and in the last section, we wrap up our paper with conclusion and future work.

2 Challenges

The main goal of a crime investigation process is to find out some indicative clues by determining the relationship between perpetrator, crime scene and the victim. As stated in Locard's Principle of Exchange *"when a person commits a crime something is always left at the scene of the crime that was not present when the person arrived."* [1]. This is absolutely true in the domain of information technologies where it is almost impossible to process something without leaving a trail behind. So what the challenging issue is to correctly determine these trails and reach the decisive data.

Digital Evidence come up with different types. While some of them are the evidence directly related with computer systems such as data files, recovered deleted files, digital photos and videos, server logs, e-mail, today, the data from a microwave oven having a computer system which indicates that the oven was programmed to arson in a specified time can also be accepted as digital evidence [2, 3].

Digital evidence, as one instance of digital data, has an abstract nature which poses a lot of problems. Evidences, by definition, must meet certain standards to be admitted legally. However, it is very difficult to conclude certain results from digital evidence having an abstract structure. The problems can be categorized under the following titles:

1. *Structural Problems:*
   - Digital data can easily be changed.
   - It is very easy to create absolutely the same digital data.
   - Digital evidence has a fragile structure and can be corrupted by some environmental factors such as magnetic field, crash or high temperature.
   - The meaning of digital data can change according to how they are interpreted or coded. Today malicious programs like viruses, worms or Trojan horse can impose different meanings to digital data.
2. *Problems originating from the Structural Problems:*
   - The Integrity of Digital Evidence: Because of that it is very easy to change, delete or replace a digital data; it is very hard to provide the integrity of digital evidence.

- The Verification of Digital Evidence: When a person is arrested with digital evidence, in the time of trial, we need to prove that the evidences belong really to that person. But captured digital evidence might have been created by someone else.
- Nonrepudiability of Digital Evidence: In the process of capturing digital evidence, the owner of digital evidence, the person handling the evidences (i.e. Police officer), the media evidences extracted and the time of capture should not be repudiated later.
- The Accuracy of Digital Evidence Data: The accuracy of the techniques, procedures used in the capture and the accuracy of the information related to the evidence (i.e. capture time information) must be proved.
- The Accountability of Digital Evidence: After the digital evidence have been collected, a third party should be able to analyze them later on.

3 Shortcomings of Digital Signatures and PKI

Today, digital signatures form the basis of a lot of online transaction used in electronic applications such as e-commerce, e-business and etc. Therefore, most of the countries go some legal regulations in their codes in order to be able to use digital signature. However, the usage of digital signature as digital evidence poses some problems and unfortunately not enough to provide a 100% certainty and accuracy in the digital evidence. Some major reasons can be listed as follows [4, 5]:

- If the private key is stolen, the whole system will be broken down. So, an adversary having gained another one's private key, can easily create signatures on behalf of the victim and in a system where digital signature is believed to provide a 100% certainty, it will be almost impossible for victim to prove his innocence.
- A private key which was stored on a file system is as secure as the security of the file system. If a security update of the system is not carried out on time when a security flaw occurred, the key can be easily compromised.
- If the private key is stored encrypted, the security of the key depends on the security of the encryption algorithm.
- The malicious codes such as viruses, Trojan horses, worms can change the way that a computer program functions without user's awareness. They have the ability to show forged data instead of the real ones. For example, in the case of an e-commerce, the malicious codes can view the total cost which is 100 000$ as 100$ by removing three zeros at the end and the user signs this with his private key without knowing that he is actually signing 100 000$ contract.
- Another important issue in PKI is how to effectively revoke a certificate. This issue has been still being discussed in the literature and no effective solution has been reached up to now [6, 7, 8]. Most of today's applications do not employ certificate revocation. Therefore, it is very hard to accept a certificate as digital evidence without knowing the validity of the certificate in the time of the action.

4 Related Work

In 2004, Maurer has proposed to use digital declarations in the domain of digital evidence in order to cope with the existing problems [9]. In digital declarations, the user performs some willful act related to the relevant contract or document, and this act is recorded digitally and combined with characteristic information of the digital document. In a typical implementation, the digital declaration can be signed together with the actual digital document i.e. recording the signer's voice digitally, declaring which amount of money he is signing and sign both this declaration and the actual electronic document together.

Beser et al. have proposed a system in 2003 in order to provide the admissibility of digital video in the time of the trial [10]. They have developed a prototype digital video authenticator (DVA) that generates digital signatures based on public key cryptography at the frame level of the digital video. Signature generation and recording is performed at the same time as digital video is being recorded by the camcorder. The DVA software is implemented on a laptop computer, which is connected to a commercial digital camcorder via the IEEE-1394 serial interface. The system also deals with the problems of PKI in following ways:

- When a case to be recorded, the key pairs are generated just before the record started and immediately after the record phase has finished, all keys are removed and not used a second time. Hence, it will be impossible for an adversary to steal the keys and later use them in other applications. The system is built on a PKI supported by the government.
- The system is located on a specialized embedded operating system which is immune to malicious codes.

These previous studies have provided us significant gains in solving some issues related with the digital evidence. However, they are still inadequate to verify all the digital evidence captured from a crime scene. So, in this paper, we aim to accomplish the following tasks:

- To build an integrated solution by utilizing the previous works and today's technology in order to be able to verify the digital evidence captured from the crime scene.
- Create this system on a public key infrastructure but obviate the problems of public key cryptography.

5 PKIDEV Model

The model which is built on public key infrastructure incorporates both the law enforcement agencies and justice mechanisms. The considerable benefit of the model is that it augments public key infrastructure with different mechanisms in order to be used in digital evidence verification. These mechanisms are EDGE, GPS and GIS Systems, Biometric Systems, Video Record Systems and Digital Declarations.

5.1 Components of the Model

The topology of the model is illustrated in figure 1 and incorporates different information and communication technologies:

EDGE (Enhanced Data rates for Global Evolution) is a GSM based technology which provides rapid wireless internet connection. In our model, it is used in real time transfer of signatures of digital evidence from the laptop in the crime scene to the central evidence servers where digital evidences will be stored.

GPS (Global Positioning System) is a satellite based navigation system, the application areas of which are mostly to detect the location of a transmitter. GIS (Geographical Information System) is used in the processes such as to process, manage, aggregate, view, analyze and model the data collected from specific geographical places. By utilizing GIS and GPS systems together, the correlation of location and graphical data is performed. In PKIDEV, we utilize these systems to detect the location where digital evidence are captured. GPS module is integrated to the laptop in the crime scene and by means of the digital map, the location of the evidences will be detected and entered into the central databases together with the other evidence information. In GPS systems, the accuracy of the detected location is directly proportional with the number of satellites. In figure 1, we have illustrated a representative model with one satellite but in order to correctly detect the location (i.e. 1m) at least 3 satellites are needed.

Digital Evidence will be stored into the central databases after they are collected in the crime scene. In our model, we have considered two centers. One of them is in police headquarter and the other is in justice headquarter. But it can be increased later. The reason why we store the evidences in more than one location is to recover the evidences when there is a problem (i.e. deliberate or accidental change or corruption in digital evidence or database errors) in one of the databases.

In our model, biometric systems are used in the identity authentication. We can call this as biometric authentication which means automatic recognition of a living being using some body characteristics. We will deploy biometric authentication with an apparatus connected to the laptop in the crime scene (see figure 1). In our framework a lot of biometric techniques can be implemented. Nevertheless, we choose the most trustable technique, fingerprinting systems.

In the course of cyber crime operation, everything will be recorded to video by the crime scene cameramen from start to end of the operation. As it is in [10], each frame of the video is signed in real time while being transferred into the computer.

In order to store the time information correctly into the database indicating when the evidences captured, it is needed a trusted time server and a signed time stamp in the PKI framework [11]. This time server should also synchronize itself continuously from a trusted party such as Greenwich.

We assume that every system that will produce a signature and every person taking an active role in combating with cyber crime and in jurisdiction of these crimes have a public and private key pair and a certificate based on PKI. We also assume that this certificate is validated in more than one point by utilizing different certificate paths in order to increase the trustability.

In the framework of PKIDEV, we also utilize digital declarations as Maurer has proposed in 2004 [9]. All the facilities related with the digital evidence in the crime

scene will also be written as declarations by hand by the crime scene detectives and afterwards these declarations will be scanned and signed with the related digital evidence together before the real time input process started. Hence, we will also be able to verify the accuracy of the evidences by using these hand-written declarations which strengthens the legal admissibility of the evidences by adding some physical nature in verification.

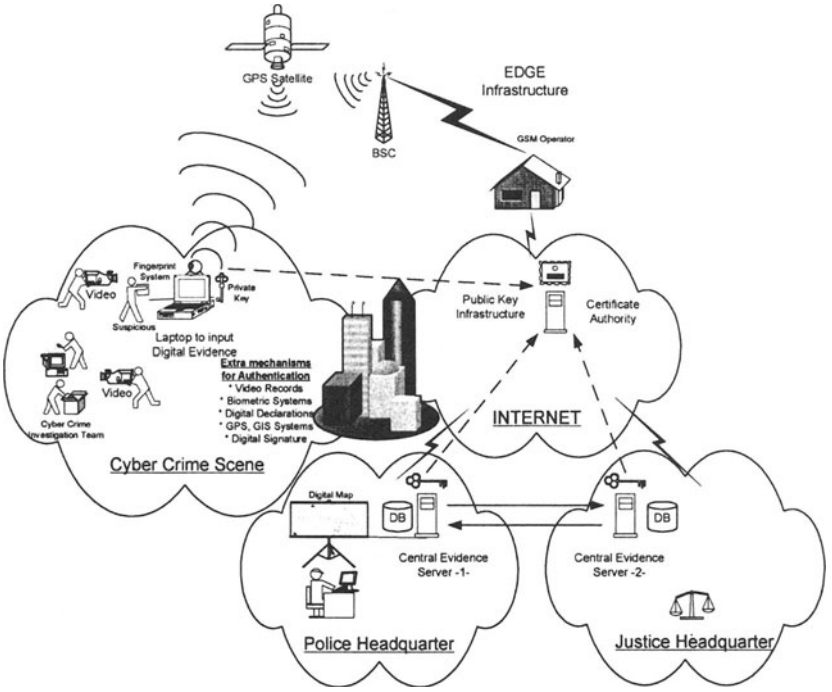

Fig. 1. PKIDEV– PKI Based Digital Evidence Verification Model

The core point in the model is the evidence input software running on the laptop in crime scene. By using the interface of this software, we can enter all data associated with the captured evidences such as the identity of the detective having seized the evidence, the type of the evidence, the specific properties of the evidence, the operation information and so on. After the input process, all the data is signed with the keys related to that evidence and signatures are sent to central evidence databases over the EDGE connection.

5.2 How PKIDEV Model Works

Before going into more detail how PKIDEV operates, it should be noted that we need a specialized and technically well equipped cyber crime scene investigation vehicle

(or more than one vehicle according to the extent of the operation) including all technical devices related to capture, store and transfer the evidences to the police headquarter without any damage.

Just after reaching the crime scene, the detective cameramen start capturing all the actions in the crime scene in the format of digital video the frames of which are signed (by using private key of the cameraman which is entered to the software just before the record started) and stored to the computer at the same time. Two different scenarios can be implemented in video capture and signing process. In the first scenario, the camcorders are connected to computers located in the vehicle with either wireless or wire lined. In the other scenario, the camcorders are connected to the mobile laptops in the crime scene. Note that it is feasible to maximize the efficiency and quality level of the records by utilizing today's high technological equipments.

After having entered the cyber crime scene, the suspicious are defused and the expert cyber crime investigation team start investigating the possible evidences. During the investigation, the fingerprints of all the suspicious are taken digitally by the fingerprinting apparatus integrated to the laptop and entered in the suspicious field of the evidence input software with the other identity information of the suspicious. After the investigation team extracted the possible evidences in the crime scene carefully, the input process is started.

The format of evidences related to a cyber crime can be handled in two categories. One of them is the digital evidence found in some kind of computer related media and the physical items including the digital evidence. So, we follow up two different methodologies in the evidence input stage.

Input of Digital Evidence: The digital evidence captured by the investigators in the course of the investigation especially the type of the evidence having a volatile structure (i.e. RAM content, the state of the network connections, open ports and etc.) are transferred carefully to the laptop. Before input started, the detective having found the evidence first gives his fingerprint to the system and than connects his smartcard to the laptop, after entering a defined username and password, input process starts. Note that the cameramen also capture the detective's input process physically (see cyber crime scene in figure 1).

Input of Physical Items: Most of the evidences are not examined in the crime scene and seized to be examined later in more suitable conditions. But the problem at this point is how to verify some issues such as the location where the evidences are collected, the collection time, the identity of the detectives having captured the evidences and etc. So, we create some digital information about the physically captured evidences and enter as an input to the system. First of all, we take photograph of each physical object (i.e. Harddisk) in a closer view. And than if it is feasible, a technical hash of each object is calculated by the help of some forensics devices brought to the crime scene. A computer forensics expert team is needed to perform all these tasks in order not to give any damage to the evidences. Again note that during these processes, the video record keeps going. One important point in recording the physical items is to zoom the special places such as serial numbers, brands etc. The digital declarations have much more importance in physically captured evidences. Before the input starts, same authentication processes as in digital evidence input stage take place. After a successful authentication, the evidence photographs, hashes and the other information found on the physical items such as serial numbers, brands etc. are entered into the

evidence input software. Evidence input software also maintains logs of every action in order to keep users responsible for their actions. Considering that these logs are very important data when a dispute is of concern, the integrity and the confidentiality of these logs should be provided. In our framework, we have used **forward integrity security property** that provides sufficient information to identify and prove any modification over the logs (for detailed information see [13]) and implemented encryption. By this way, any alteration, deletion on these logs can be detected and only the authorized persons can see the content of them.

After input stage is completed by clicking to submit button, system first connects to the defined time server securely and gets the current time. Than all the input data (the attributes of the evidence, the related operation, the identity of the detectives etc.) is combined with time information, fingerprints of the detectives and digitalized physical data (photographs, video records, and digital declarations). All of them are signed with the private key of the detective found in the smart card[1]. At first glance, to store all these gathered evidence data and signatures into the centers in real time might be considered. But because of the fact that our collected data includes video and other high volume data and the limitations of EDGE bandwidth it is impractical. Hence we have preferred to send only the signatures of the data and the metadata not the evidence itself in real time. The original evidences will be transferred to police headquarter in out-of-band channel and than stored in both databases. Note that there is a fast channel i.e. fiber optics between two databases which are in police and justice headquarters.

In the police headquarter, when the signatures are received, the location information is detected by GPS (Location data is also entered to the software and also stated in the digital declarations) and entered into the central evidence server found both in the police headquarter and in the justice headquarter together with an incident number assigned automatically by the system.

At the end of the operation, all the evidences (physical and digital) are packed, stamped and transferred to the computer forensics center in convenient environment conditions. In this center, similar methodologies as in the crime scene are followed, when new evidences are explored.

Another crucial issue is the security of the model itself. For example, the security of central databases, used software programs, operating systems, methods, procedures and cryptographic algorithms must definitely be taken into consideration.

6 Benefits of the Model

In PKIDEV, authentication in depth approach is followed, thus resulting in strong verification of who has collected the evidences and from whom these evidences were collected. For example, in the crime scene, the fingerprints of the suspicious were entered the system, while this action was also being recorded to the video at the same time. By using the detective's public key, the assurance that the associated evidence

[1] In our scenario, in order to increase security, we perform signing process in the smart card without revealing the private key [12].

information has not been altered since it was obtained will have been provided. Evidence information also includes the identity of the suspicious. Therefore, it will be almost impossible that the suspicious denies his availability in the crime scene. The detective who has handled the evidences will be able to be authenticated by the help of his fingerprint, his public key, video records, his digitalized hand-written declaration and the evidence input software logs on which forward integrity is used [13]. Under these conditions, it will be also impossible for the detectives to deny their availability in the action.

The location where the evidences are captured is automatically detected by GPS and GIS systems. Furthermore, it can also be reached from the declarations and the evidence input software records.

The time information indicating the capture time of the evidences is received from a time server securely and automatically appended to the evidence data in the time of the input. And also it can be reached by the other records such as digital declarations, evidence input software data, video records and etc.

In our model, all the digital evidence captured from the crime scene, are signed and these signatures are stored to different databases. Physical items that may contain potential digital evidence are hashed and signed by the help of some forensics devices and the other directly captured digital evidence are signed in the time of the input. Moreover, all the activities including this signing phase are recorded to video and also all the details are written as declarations. Since, the integrity will be physically validated as well as digitally, the denial of these evidences will not make so much sense in a court of law.

One other significant point in digital evidencing is to prove the accuracy of techniques, knowledge, programs, methods and procedures used in the course of capturing the digital evidence. We assume that every component in our model is certified by a world-wide known, trusted and accepted organization.

In our model, all the evidences and the data about them are stored in two different locations together with their signatures. One of them is police headquarter and the other is justice headquarter. if all the processes and authentication mechanisms that we have mentioned up to now is implemented correctly, there will be no reason that a third party can not analyze the evidences later. This provides the accountability of the digital evidence. Furthermore, because of that evidences are stored in more than one location; the claim that one party has altered the evidence can easily be refuted.

7 Conclusion and Future Work

In this paper, we have presented a PKI based digital evidence verification model which is called PKIDEV in order to provide security for the process of capturing and preserving digital evidence collected from a crime scene.

PKIDEV can also be regarded as the first proposed integrated model trying to solve the verification problem of the digital evidence captured from a crime scene in order to maximize legal admissibility in the time of trial.

PKIDEV combines a lot of different cryptographic techniques with the latest technologies such as GPS, EDGE and implements authentication in more than one level.

In our study, in order to cope with some existing problems public-key cryptography brings when it is applied to the verification of digital evidence, we have incorporated digitalized physical objects, such as scanned hand-written declarations and video records in the authentication phase as well as the other well known authentication techniques such as digital signatures, secure time-stamping and etc. To preserve the integrity and the availability of the evidence data, we have stored the same evidence data in more than one location.

The digital evidence related to a cyber crime are not only limited with the crime scene. A lot of digital evidence can be found (especially from the internet) without going to crime scene. In this paper, we have not argued to verify all the digital evidence but we have focused on a more specific case which is to verify only the digital evidence captured from the crime scene. So, a promising future work might be to implement similar scenarios to the other cases i.e. to verify the internet audit logs of internet service providers as source of digital evidence or to provide the integrity of digital evidence captured from the internet in real time.

With PKIDEV, we have actually tried to build an initial model for the verification of the digital evidence. The model has a wide topology and incorporates a lot of different authentication techniques. So, as a future work, we believe that PKIDEV can be extended or reviewed in many points to increase the security or the efficiency of the verification processes.

References

1. Casey, E., Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, 2nd Ed., Academic Press, 2004
2. Shinder, D. L., Scene of Cybercrime, USA, 2002
3. Uzunay, Y., Koçak M., "Digital Evidences in the Domain of Cyber Crime", AB'05 Academic Informatics Conference, Gaziantep, Turkey, February 2005
4. Caloyannides, M.A., "Digital Evidence and Reasonable Doubt", IEEE Comp.Society, 2003
5. Oppliger,R., Rytz, R.,"Digital Evidence: Dream and Reality", IEEE Computer Society, 2003
6. Kocher, P., "A quick introduction to certificate revocation trees (crts)", 2000
7. Myers, M., Ankney, R., Malpani, A., Galperin, S., Adams, C., "X.509 internet public key infrastructure – online certificate status protocol - ocsp", IETF, RFC 2560, June 1999
8. Bicakci, K., Crispo, B., Tanenbaum, A.S., "How to Incorporate Revocation Status Information into the Trust Metrics for Public-Key Certification", SAC'05, ACM, March 2005
9. Maurer, U., "New Approaches to Digital Evidence", Proceedings of the IEEE Vol. 92, No.6 June 2004
10. Beser, N.D., Duerr, T.E., Staisiunas, G.P., "Authentication of digital video evidence", SPIE, San Diego, California, August, 2003
11. Hosmer, C., "Proving the Integrity of Digital Evidence with Time", IJDE, Vol.1, Spring 2002
12. Fratto, M., "Security Tokens", August, 2001
13. Schneier, B., Kelsey, J., "Secure audit logs to support computer forensics". ACM Transactions on Information and System Security (TISSEC), 1999

[i] Supported by OYP

Professionalism in Computer Forensics

Alastair D. Irons[1] and Anastasia Konstadopoulou[2]
[1]School of Computing, Engineering and Information Sciences
Northumbria University
2.12 Pandon Building
Camden Street
Newcastle
NE2 1XE
United Kingdom
E-mail alastair.irons@northumbria.ac.uk

[2]Department of Computing
University of Bradford
Bradford
BD7 1DP
United Kingdom
E-mail a.konstadopoulou@bradford.ac.uk

Abstract
The paper seeks to address the need to consider issues regarding professionalism in computer forensics in order to allow the discipline to develop and to ensure the credibility of the discipline from the differing perspectives of practitioners, the criminal justice system and in the eyes of the public. There is a need to examine and develop professionalism in computer forensics in order to promote the discipline and maintain the credibility of the discipline.

The paper explores the hallmarks of a profession using Denning's (2001) criteria and applies these hallmarks to computer forensics. The paper attempts to determine the position of the discipline in relation to other forensic science areas and in relation to computer science. The paper seeks to identify professional issues and challenges for computer forensics and links these challenges to legal and ethical considerations. Consideration is also given to issues such as post-traumatic stress disorder.

The issue of certification of practitioners is raised and questions regarding who should certify and what they should be certifying are discussed. The certification issues are of course related to the position of the discipline but are also central to the credibility of the discipline and the ability to ensure robust and due process when computer forensics is applied to the criminal justice system and other disciplinary systems. The role universities have in developing the subject of computer forensics is also considered.

A very draft version of a practitioner framework is introduced. This is the subject of current work being developed and seeks to take forward the issues raise in this paper as the basis for future certification and accreditation of computer forensics practitioners.

Keywords professionalism, accreditation, certification, credibility, standards, education

1. Introduction

There is a growth in the number of criminal and industrial investigations making use of computer forensics, in the number of organisations offering computer forensic services and in the number of computer forensics practitioners. As a result of this growth there is a need to consider professional issues in order to legitimise the subject and develop a professional framework which will assure confidence from the public, from the criminal justice system and from all organisations using computers or computer systems.

The paper will utilise Denning's (2001) hallmarks of a profession in the discussion of professionalism in the computer forensics domain, namely that a profession should contain:

- a durable domain of human concerns;
- a codified body of principles (conceptual knowledge);
- a codified body of practices (embodied knowledge including competence); and
- standards for competence, ethics and practice.

It is important to examine professionalism in the context of computer forensics in order to consider the

- expectations of computer forensics professionals;
- amount of knowledge and skill computer forensics professionals require;
- ability computer forensics professionals to resolve computer forensics investigations and problems;
- ability of computer forensics professionals to design the tools, techniques and procedures which will enhance computer forensics as a discipline.

The computer forensics professionalism in this paper focuses on the investigative computer forensics practitioner rather than software or hardware developments associated with working in the computer forensics subject area or technical sales representatives working on selling computer forensics products.

One of the main challenges in considering professionalism in the context of computer forensics is that computer forensics is a relatively new discipline, certainly when compared to established disciplines in forensic science or computer science. There is a question as to whether computer forensics should be part of the ongoing general debate on professionalism in computing – remembering that computing is a very new discipline in its own right. Computer forensics as a discipline is further complicated by the speed of technological change and the change in the activities of computer criminals and other computer "misusers".

It is suggested in this paper that computer forensics professionals should act with honesty and integrity, be accountable for their actions and have appropriate technical competence – evidenced by appropriate qualifications and professional body membership. It is also suggested in this paper that it is incumbent upon computer forensics professionals to participate in the following professional duties and responsibilities:

- promote computer forensics as a discipline, including the promotion of appropriate professional bodies for computer forensics;
- participate in continuous personal professional development;
- develop of tools and techniques which will enhance the discipline, help the legal process and reduce computer crime;
- raise awareness of the benefits to business and society afforded by computer forensics; and
- participate in creating operational processes and procedures to counter computer misuse and digital crime.

2. Professionalism in Computing

Organisations such as the British Computer Society (BCS) proscribe to the notion that the creation, development, management, utilisation and maintenance of computer systems is a professional activity in which qualified computing and IT professionals are recognised and respected. There is a perception that through professionalism in computing and IT that the quality of computer and information systems will be improved. However computing has a number of significant professional issues focussing on:

- being recognised as a profession in its own right – with competent and qualified practitioners who have relevant up-to-date skills, qualifications and experience and are committed to an ongoing process of Continued Professional Development (CPD);
- entry into the profession – very few people working in computing require a formal qualification as a condition of that employment.;
- practitioners working to a code of ethics showing personal integrity, responsibility and accountability;
- failure to deliver computer systems on time and on cost;
- failure of systems when they are in a live environment – and the consequent negative impact on business and society

It is not the purpose in this paper to examine in detail the many professional issues facing computing but to recognise that there are issues which need to be resolved in order to enhance the perception of computing from all stakeholders (government, business, industry, employers, users and the public). Many of the professional issues facing computing are also relevant to professionalism in computer forensics.

3. Positioning Computer Forensics

There are a number of professional bodies and organisations which potentially have an interest or stake in computer forensics. However, at present such professional bodies do not have the power to award a licence to practice in computer forensics and therefore are not able to able to prevent unqualified or under qualified people carrying out computer forensics work. More importantly the professional bodies are not able to stop people from offering computer forensics services or practising computer forensics and are unable to bar people from the profession as a result of unprofessional practice or misconduct.

It could be argued that the market place could be able to control the levels of expertise in computer forensics, but it is argued here that the market place do not have the awareness or ability to determine acceptable computer forensics skill levels. Added to that is the shortage of people with appropriate computer forensics skills and

organisations are willing to employ people who do not have the requisite computer forensics skills and competences.

Another controlling factor may be attributed to the legal system and it would be hoped that any cases going to court which had not engaged computer forensics specialists with appropriate skills would be thrown out. However, this depends on the effectiveness and the understanding of the judicial systems. In many cases the damage will already have been done by the time either a case gets to court and fails or is stopped before it gets to court as a result of inappropriate (or unprofessional) computer forensics practice.

There is an ethical aspect to professionalism in computer forensics in that computer forensics practitioners may find themselves in positions where they have to make choices, these could be technical, procedural or actual ethical dilemmas – but all could have ethical impacts on particular cases. There will often be an element of power in computer forensics investigations which will potentially have an ethical impact on any decisions made.

In considering professionalism of computer forensics there is a need to determine which professional body that computer forensics professionals should align with, whether to align with forensic science or computer science or to have a professional body focussing solely on computer forensics.

Forensic science – for example aligning with forensics science organisations such as the Council for Registration of Forensics Practitioners (CRFP). The British Computer Society (BCS) were consulted on CRFP's proposals to register digital evidence specialists giving support to the proposed link. Linking with the CRFP would have the benefit of aligning with other forensic science practitioners, but would not necessarily take into account the different (from other forms of evidence) nature of digital evidence (virtual, non-static and volatile) or the rapid and continual changes in computer technology;

Computer science – working with professional bodies in computing such as the British Computer Society (BCS) or Association of Computing Machinery (ACM). These professional bodies are beginning to move towards specialist certification; for example, BCS have advised on forensic practice on digital evidence. However, as McKemmish (1999) suggests computer forensics is different from other areas of computing in that there is a requirement that the final result from computer forensics investigations legally acceptable;

Computer Forensics Body – Rogers (2003) has argued that there is a growing common body of knowledge which establishes computer forensics as a unique area of study. If the discipline establishes itself as a unique discipline then a professional body dedicated to developing codes of practice and ethics for computer forensics practitioners and involved in accreditation and certification of computer forensics practitioners may be appropriate. The ACPO Guidelines go some way to addressing standards but do not determine regulations for practice. Other organisations such as the International Association of Computer Investigative Specialists (IACIS) and the High Technology Crime Investigation Association (HTCIA) focus on the standards for investigative computer forensics practitioners, mainly in law enforcement. The Institute for Computer Forensics Practitioners (ICFP) was established in 2004 to create a "new standardisation, education and foundation of principles and practices in digital forensics that would be open to both public and private sector practitioners", (Schroader and Dudley-Gough, 2006) – see later discussion on certification.

Not only the identification of the discipline and where it lies, but also the nature of tasks involved in computer forensics. As well as skills development there is the continuous change of the discipline; the changing nature of computer crime; the continuous need to develop new tools and techniques and the changing legal environment which need to be taken into account in order that computer forensics has credibility and is accepted by the criminal justice system and by the public.

4. Professional Issues in Computer Forensics

Whilst many of the issues that apply to professionalism in computing also apply to professionalism in computer forensics there are a different set of professional issues and values which arise due to the nature of computer forensics, forensics investigations and the analysis of digital evidence. In simplest terms, if a professional approach is not used in computer forensics investigations then the investigation and the subsequent prosecution will probably fail. There are specific professional issues in computer forensics including, evidential integrity, evidential continuity and trans-jurisdictional cases. Practitioners in computer forensics may also find themselves in positions of power (not always sought after power) and may find themselves subject to temptation to get involved in computer crime themselves – hence the need for background checks on computer forensics practitioners.

Related to professional issues in computer forensics are the particular legal and ethical issues associated with the subject. In order to address professional certification there is a need to take into account the legal and ethical perspectives as well as the technical requirements. As with the discipline in general, whilst many of the legal and ethical issues associated with computing and computer science can be applied to computer forensics there are also a series of unique legal and ethical considerations which need to be taken into account. The consideration of specific legal and ethical issues pertaining to computer forensics is the subject of current research being undertaken by the authors.

As well as having a central body which certifies and accredits practitioners before they practice computer forensics which also has the power to stop practitioners from practicing there are a number of issues which should be considered in examining computer forensics as a profession. There will be some who speak against a central body citing standardisation as an unnecessary constraint and worrying that too much control will lead to computer forensics declining as a discipline.

However if Denning's (2001) hallmarks of a profession are considered, it can be seen that computer forensics is moving towards a profession in its own right – to this end computer forensics can draw principles and practices from both forensic science and computer science as well as initiate and innovate in developing principles and practices specific to computer forensics.

1 A durable domain of human concerns

There is a concern with computer crime and the impact that it has on society both at an individual level and for society as a whole. Computer forensics as a discipline can be used to reassure the public that those involved in carrying out computer crime will be dealt with and society will be a safer place as a result. It is also hoped that computer forensics, along with the related discipline of computer security, will act as

a deterrent to computer crime – by ensuring that computer criminals are caught and by having in place mechanisms to catch them.

2 *A codified body of principles (conceptual knowledge)*

Computer forensics is moving towards a core body of knowledge (Rogers, 2003). In its simplest terms computer forensics is, according to Bates (1997), "the scientific examination and analysis of data held on or retrieved from computer storage media for the purposes of presentation in a court of law, together with the study of the legal aspects of computer use and misuse", and to that extent there is a shared set of goals which will eventually reduce computer crime and be to the benefit of the public. On the other hand a different perspective of computer forensics centres on making money for organisations, reducing labour costs and mechanising computer forensics investigations which might not be so altruistic in benefiting the public good.

3 *A codified body of practices (embodied knowledge including competence)*

The ACPO principles are an example of guidelines for practice which identify specific methods and expectations for computer forensics investigations. On the other hand, the plethora of hardware potentially involved in computer forensics investigations, the variety of operating systems and the spectrum from stand alone PCs to global networks mean that common methods will by their nature be at a generic level. However, the acceptance of common principles, such as evidential integrity and evidential continuity, mean that even in cross-jurisdictional cases there needs to be commonality in approach.

4 *Standards for competence, ethics and practice*

Common standards are required in computer forensics to ensure consistent approaches in

- obtaining digital evidence;
- ensuring it is the digital evidence that the investigation is seeking; and
- analysing the digital evidence once it has been obtained.

In order to manage standards in a profession the entry to membership of that profession requires extensive formal education, not just practical training or apprenticeship.

An interesting aspect of computer forensics is that because of the rise in computer crime and the demand for people to work in computer forensics people may be allowed into the profession without formal education. A cynical view would be that many computer forensics practitioners offer computer forensics services because they see the opportunity to make money rather than to provide a service which makes society safer. Everett (2005) estimates that up to 20% of practitioners involved in computer forensics activities are not competent or appropriately certified.

There is much to do in standards for competence (ICFP are beginning to attempt to address this) and the development of an ethical code of practice (ICFP has established a code of ethics focussing on integrity, impartiality, diligence and objectivity – but without going into each of these in pragmatic detail). In order for a professional body in computer forensics to be effective it needs to have the power to register or certify computer forensics practitioners before they are allowed to practice and has to have

the power to bar from practice when there have been breaches of code or instances of incompetence. Until this is in place there will remain issues in assuring standards.

In order for a profession to manage its own standards of practice and ethics a degree of self regulation is required. Self-regulation in a profession requires agreement (either tacit or implied) that the public allows special societies or organisations that are controlled by members of the profession (self regulation). These societies are responsible for setting the standards for admission to the profession, setting the standards of conduct for members, and have the power to enforce those standards. One of the challenges for computer forensics is to define the professional body most relevant for computer forensics.

5. Certification in Computer Forensics

Attempts have been made to introduce licence to practice procedures in Computing and Software Engineering, for example in Texas (Bagert, 2002) in order to enhance the standing of the profession, although questions have been raised about the benefits of certifying computing practitioners and software engineers (Knight and Leveson, 2002). In computer forensics the debate potentially takes on a new lease of life and there is the need to have discussion on whether standards can be defined that a licensed practitioner would be obliged to adhere to, issues around the provision of legal evidence and practitioner issues around acting as an expert witness. Jones (2004) argued that "there is an absence of standards and competencies in the field of cybercrime" and although the situation has progressed in the last two years there is not a unified position in certification and accreditation.

There are a number of certifications available to the computer forensics practitioner including qualifications primarily aimed at computer security professionals such as Certified Information Systems Security Professional (CISSP) from $(ISC)^2$ – which focus, as one would expect on the computer security rather than computer forensics. A number of specific computer forensics certifications are beginning to appear such as the Cyber Security Forensics Analyst (CSFA) from the Cyber Security Institute. Interestingly this qualification requires participants to complete a comprehensive practical examination, but it is also recommended that exam participants have 18 months experience before they attempt the exam. There are also certifications from commercial organisations for specific products such as EnCase from Guidance Software® have produced certification for the EnCase product in the form of the EnCase Certified Examiner (EnCE) certification.

In order to develop the autonomous standing of computer forensics consideration should be given to professional body alignment and licence to practice. There is an expectation, as stated in the ACPO (2003) principles, principle number 2, regarding the competence of practitioners in being able to undertake computer forensic activities; "in exceptional circumstances, where a person finds it necessary to access original data held on a computer or storage media, that person must be competent to do so and be able to give evidence explaining the relevance and the implications of their actions". Although the principle alludes to exceptional circumstances the implicit expectation is that computer forensics practitioners should be competent before undertaking any computer forensic duties.

The Council for the Registration of Forensics Practitioners (CRFP) has begun to allow computer forensics practitioners to apply for registration (Everett, 2005) but

there remains a question as to which is the most appropriate accrediting body for computer forensics practitioners.

A significant factor which differentiates computer forensics professionals from other computing professionals is the possibility of acting as an expert witness and the ability to act as an expert witness may require further, and separate, certification and accreditation. "Courts in the UK are taking an increasingly tough stance on expert witnesses who do not comply with their duties to the Court" according to Ellison (2005) and it may be, along with expert witnesses in other disciplines, computer forensics experts will be required to show evidence of their competence to act as an expert witness.

6. Continuous Professional Conduct

Even in the case of establishing the most appropriate accrediting body, obtaining registration with the organisation should only be the first step of establishing the professionalism of the member. Maintaining the professional conduct is an ongoing process and evidence of its ongoing process should be continuously sought by the accrediting body and should include evidence of ethical conduit, and standards of behaviour. This process should include evidence of exhibiting the highest level of ethical behavior at all times and maintaining objectivity and confidentiality during an investigation. The practitioners have amoral responsibility to enrich their technical knowledge the subject and conduct the investigations with integrity.

The cross-disciplinary nature of computer forensics is significant, requiring the linkage between investigatory techniques, computing technologies and jurisprudence. Britz (2004) suggests that "it is essential that the potentiality of computer-related crime and the insidious nature of the phenomenon be recognized and addressed by all sectors of the community"

7. Consideration of Post Traumatic Stress Disorder

Although this section may seem somewhat out of context with the rest of the paper – there is a very real professional issue in dealing with the stresses associated with working on computer forensics cases. It is important to realise that professionals working in computer forensics will come across cases which have added professional pressures, such as dealing with cases which will potentially have a psychological impact on their work and their non-working life – for example investigating paedophile or other obscene material cases. Constant exposure to such material may lead to the potential desensitising of the practitioner to the obscenity of such material or even lead to practitioners suffering symptoms similar to "post traumatic stress disorder".

In order to counter potential post-traumatic stress disorder in computer forensics professional one possible solution is for each practitioner to have an independent professional confidentiality but at the same time providing an outlet for discussion. The mentor would be a professional person who would have a similar job and who;

- would provide an outlet to share experience and concerns with;
- had the ability and willingness to provide independent guidance and advice;
- had the skills to act as a critical friend;
- would and could help plan and discuss career and professional development.

8. Practitioner Framework for Professionalism in Computer Forensics

In this section a practitioner framework is outlined. This framework is currently in development and is aimed at providing a practitioner framework for computer forensics professionals. The framework will be able to be used in considering the abilities of practitioners and their suitability to be involved in particular cases and procedures (building on the ACPO Guidelines for Handling Digital Evidence). The framework will also include the duties and responsibilities outlined in the introduction of this paper. It is planned to develop this framework in the near future.

The headings in the proposed framework will include

- Qualifications;
- Professional body membership;
- Accreditation;
- Certification;
- Consideration of legal, ethical and social concerns;
- Continued Professional Development;
- Reflective Review of Practice; and
- Promotion of the Discipline

9. What can Universities do?

Universities are able to contribute to the enhancement of professionalism in computer forensics by raising awareness and promoting professionalism in computer forensics. Universities have an important role in promoting computer forensics in the public domain and raising awareness about the strengths and issues associated with computer forensics and indeed computer crime.

Universities are in a position to provide education in computer forensics through the provision of undergraduate and post graduate programmes which cover the principles and techniques of computer forensics and are accredited by relevant professional bodies and give graduates certification in computer forensics. In the development of such programmes it is important that they obtain external validation and verification from independent bodies (for example from BCS or CRFP) in order to ensure standards, and to address the issues of accreditation and certification of computer forensics practitioners. Universities may also be able to work in collaboration with organisations such as Guidance Software in embedding the ENCE examination into the curriculum (similar to the relationship many universities currently have with organisations such as CISCO). It is important that universities work in collaboration with professional bodies, practitioners, police forces and legal experts to provide programmes which will address the needs of the industry – in the provision of suitably skilled and certified experts.

In order to develop and sustain computer forensics as a discipline in its own right, the discipline needs to be underpinned by relevant research. Universities are in a position to take forward the computer forensics research agenda (assuming funding is available) and can make a valuable contribution to the disciple by using expertise to undertake relevant research for the discipline. It is suggested in this paper that as well as theoretical and 'blue sky' research opportunities universities should be involved in

applied research in computer forensics and that this applied research should be in collaboration with computer forensics practitioners and organisations in the public and in the private sector.

As well as providing opportunities for education, universities are able to provide a co-ordinating role in taking the discipline forward. There is an important role for academia in both research to assist computer forensics practitioners, and in educating and preparing future computer forensics specialists. It is suggested in this paper that universities should attempt to work together as a consortium – not in competition with each other – to develop the common body of knowledge and enhance the subject through collaborative teaching and research.

Universities have the opportunity to provide a balanced and multidisciplinary view on the subject of computer forensics. There is a potential danger that we move forward on the investigation and countering crime agenda and fail to give appropriate consideration to the rights of individuals and the protection of society. Universities can provide different perspectives on the relative merits of computer forensics and thus counter the possibility of promoting computer forensics as a set of activities that do not act to the benefit of society.

10. Future Development

The development of computer forensics as a discipline is likely to continue, at the very least in the short term. Technical developments in operational procedures, in technology and computer forensics tools will be required in order to keep pace with developments in cybercrime. However, parallel developments in professional responsibility, certification and accreditation are required in order to maintain standards and the standing of the discipline.

11. Summary

This paper has attempted to raise a number of the issues associated with professionalism, certification and accreditation in computer forensics. It is suggested that in order to maintain the credibility of computer forensics as a discipline there is a need to raise the importance, and address the requirements, of professional issues in computer forensics.

It is advocated that computer forensics practitioners will require suitable certification and accreditation in the future. The body which manages this process will need to embrace principles of self regulation as well as having the power and authority to provide licence to practice and to bar from practice when required.

There is a need to address and formalise the way the industry deals with professional and certification issues. It is suggested that on order to take this agenda forward that professional bodies, practitioners, stakeholders in the criminal justice system and universities should work together to produce a workable and manageable solution.

References

Association of Chief Police Officers (2003) *Good Practice Guide for Computer Based Electronic Evidence,* NHTCU, London

Bagert, D. J. (2002) 'Texas licensing of software engineers: all's quiet for now' *Communications of the ACM*, Vol 45, No. 11

Bates, J., (1997) 'Fundamentals of computer forensics' in *International Journal of Forensic Computing* [online] accessed December 2005

Britz, M.T., (2004), Computer Forensics and Cyber Crime, Pearson Prentice Hall, New Jersey

Denning , P., (2001) 'The profession of IT: who are we ?' in *Communications of the ACM*, Vol 44, No. 2, pp 15 – 19

Ellison, J., (2005) 'The importance of being earnest – toughening up on experts', in *Forensic Accountant*, Issue 28, summer 2005, pp 2 – 3

Everett, C., (2005) 'Forensics – cred or crud', in *Digital Investigation*, Vol. 2, No 4., pp 237 - 238

Jones, N., (2004) 'Training and accreditation – who are the experts?' in *Digital Investigation*, Vol. 1, No. 3 pp 189 – 194

Knight, J.C., and Leveson, N. G., (2002) 'Should software engineers be licensed' in *Communications of the ACM*, Vol. 45, No 11, pp 87 – 90

McKemmish, R., (1999) ' What is forensic computing ?', *Trends and Issues in Crime and Criminal Justice*, Vol. 118, pp 1 – 6, Australian Institute of Criminology, Canberra

Rogers, M.K., (2003) 'Computer forensics: science or fad', in *Security Wire Digest*, Vol 5, No 65

Schroader, A., and Dudley-Gough, N., (2006) The Institute of Computer Forensics Professionals, in *Digital Investigation*, Vol. 3, No 1, pp 9 – 10.

Smith, F. C. and Bace, R. G., (2003), 'A Guide to Forensic Testimony, The Art and Practice of Presenting Testimony as an Expert Technical Witness'. Addison-Wesley, Boston, USA