Cracking Passwords in Forensic Investigations: Cost Implications

Vishal Bulland
Bachelor of Computer Science (Pune University, India)

a thesis submitted to the graduate faculty of design and creative technologies
AUT University
in partial fulfilment of the
requirements for the degree of
Master of Forensic I.T.

School of Computing and Mathematical Sciences

Auckland, New Zealand
2010

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a University or other institution of higher learning, except where due acknowledgement is made in the acknowledgements.

....

Signature

Acknowledgements

This thesis was completed at the Faculty of Design and Creative Technologies at AUT University in Auckland, New Zealand. While conducting research I received a lot of valuable support from many people in one way or another. Without all of this help and support, it would not have been possible for this thesis to exist in its current form. I would like to take this opportunity to deeply thank all of you for your kind support. I would like to apologise to those whose names were not mentioned here; however, I highly valued your support.

Firstly, I would like to deeply thank my supervisor, Dr Brian Cusack. Dr Brian Cusack has helped in many uncountable ways. To begin with, I am thankful for his important decision to choose to supervise me. Dr Cusack also gave me a lot of advice and support in order for me to explore and choose the direction of research I was interested in pursuing. I thoroughly enjoyed his daily supervision and my discussion sessions with him have been very fruitful. I have gained a great deal of knowledge just from my discussion sessions with Dr Cusack. Not only has Dr Cusack provided me with very valuable guidance, but he has also been a very kind an encouraging motivator. Dr Cusack has helped me to shape the thesis and has also actively supported and motivated me in my endeavours to perform the best possible research.

I would also like to thank my lecturer Mr Campbell McKenzie. Mr McKenzie has also helped me and provided me with the knowledge required in the field of digital forensic investigations. Mr McKenzie, being a digital forensic investigator himself, was able to give me a lot of valuable information and guidance with regards to best practices followed in the industry. The guidance provided by Mr McKenzie was not only priceless but it also helped me a great deal to perform this research study.

Next, I would like to thank Mr David Lewis, director of Fulcrum Management. He was kind enough to provide our research labs with the necessary software. Mr David Lewis has helped both the AUT Digital forensic research laboratories and me by providing us with AccessData's Distributed Network Attack software suite. I would also like to thank him and AccessData for the excellent product support.

I would also like to deeply thank Prof Neil Binnie for his advice and guidance. Prof Neil Binnie also helped me in my research by providing me with valuable guidance. Prof Neil Binnie was kind enough to actively support me and give me valuable knowledge to perform my research analysis and discussions.

I would also like to thank my fellow Master of Forensic I.T. classmates. I have learnt a great deal just from discussing all study matters with them. I have also gained a lot of moral support and encouragement from their company and friendship.

I would also like to thank my editor Marjorie Carlson. She helped me a great deal by editing and proofreading this thesis in order to make it reader friendly.

I would like to thank all of my family members. Without their love and kind support, I would not have been able to perform research. I would like to thank my mother, Mrs Pooja Bulland, and my father, Mr Ved Bulland, for the values that I learnt from them, which made me who I am in my life today. I would like to thank my younger brother, Dhiren Bulland, for his love, kind support, and company during my research study here in Auckland. My family means the world to me, and the research that I have performed is all a result of their endless love and kind support.

Finally, a big thank you to everyone responsible in any way, to those whose names were mentioned here and also to those whose names I accidentally forgot to mention and give due credit to.

Abstract

Digital Forensic Investigators need to forensically analyse digital data in order to investigate various crime cases. Quite often, the investigators come across password protection for the digital data that they need to investigate. Therefore, they need to crack passwords in order to gain access to potential evidence. There exist various problems in the field of password cracking. Due to technological advances in security, it currently takes—and is expected to continue taking—large amounts of time for digital forensic investigators to crack passwords. Due to the large amount of time required to crack passwords, the costs involved in cracking the passwords are also high. There also exists an ambiguity with regards to the outcome of the password-cracking procedure. Thus, there is a risk of the forensic investigator not being able to find the correct password even after spending large amounts of time and money. Apart from the ambiguity of the outcome and large password-cracking times, there also exists uncertainty regarding the amount of time a password may take to crack. While a variety of research in the field of password cracking exists, past studies have mostly concentrated on the underlying password-cracking technology in use. They have not examined the underlying procedures and practices involved in cracking passwords.

In order to address the various challenges mentioned above, this research proposes the use of a budgeting model. The budgeting model aims to gain control over the amount of time required to crack a password. This also makes it possible to gain control over password-cracking costs. This research also defines an experimental design to define and test the processes involved using the budgeting model. This research consists of a simulation of 200 hypothetical password-cracking cases, classified in groups or blocks of 50 password cases. The various time budgets for each block of passwords are calculated before the actual password-cracking experiment is performed. The password-cracking experiment is then performed as per the defined processes for a period of seven days. The experiment is also monitored regularly. The actual password-cracking times for all of the passwords are also recorded.

The data are then analysed. There are certain variations involved in the processes and results, which have been considered during analysis. The results find that the actual password-cracking times were less than the times allocated by the use of the budgeting model. Therefore, the budgeting model guidelines are demonstrated suitable to be followed as best-practice advice for digital forensic investigators. The results also show that the actual times required to crack the passwords are very near to half of the expected budgeted time.

This suggests that, on average, the password-cracking times are half of the required budgeted time. The various research processes carried out are also evaluated in order to add to the existing best-practice knowledge for digital forensic investigators. Based on the findings of this study, the recommended budgeting procedure for digital forensic investigators is also outlined.

Table of Contents

Chapter 1 - Introduction

Chapter 2 - Literature Review

Chapter 3 - Research Methodology

Chapter 4 - Research Findings

Chapter 5 - Research Discussion

Chapter 6 - Conclusion

List of Tables

List of Figures

List of Abbreviations

AMD	Advanced micro devices
ANSI	American National Standards Institute
BIOS	Basic Input/Output System
CMOS	Complimentary metal-oxide semiconductor
DES	Data Encryption Standard
DNA	Distributed Network Attack
FPGA	Field programmable gate array
GPU	Graphics processing unit
HDD	Hard disk drive
LANMAN	Lan Manager
MCF	Modular Crypt Format
MD5	Message Digest 5
MPI	Message Passing Interface
PCFG	Probabilistic Context-Free Grammars
PKCS	Public-key cryptography standards
PRTK	Password Recovery Toolkit
PS3	Playstation 3
RSA	Rivest, Shamir and Adleman
TACC	Tableau TACC1441 Hardware accelerator
XMPP	Extensible messaging and presence protocol

Chapter One

INTRODUCTION

1.0 BACKGROUND

The use of passwords for securing data is a common practice for computer users. Passwords are used as a security mechanism to protect the privacy and confidentiality of data on computers. Passwords are used for a variety of applications, for example for securing a person's e-mail account, bank account, computer, or private data in files and spreadsheets. Password protection not only ensures authentication of the user accessing the data, but in many cases it also supports encryption for the entity that needs to be protected. Encryption ensures the privacy of data, such that only the legitimate user can have access to the information. Therefore, any illegitimate user who does not have the password is unable to have access to the encrypted information.

Passwords have proved to be optimal for security. However, passwords can also be used by people with malicious intent. A malicious person may use passwords and encryption to protect incriminating information from the access of law enforcement agencies. If the law enforcement agencies or forensic investigators are not provided with the password, it may not be possible for them to gain important evidence. Such a situation could hamper justice, as the malicious person may not be prosecuted in a court of law without credible evidence. Therefore, in such situations law enforcement agencies need to resort to various means of password recovery in order to gain access to the encrypted information.

One of the most common methods of overcoming password protection is password cracking (Casey, 2004). Password cracking can be done using a variety of tools and techniques, the most common of which is to use automated software to try various possible guesses until the correct password is found. Other techniques involve the use of brute-force cracking methods, rainbow tables, and also Markov chains (Marechal, 2008). Hence, in order to recover passwords, law enforcement agencies need to use a variety of password cracking tools and techniques.

1.1 PROBLEMS FOR DIGITAL FORENSIC INVESTIGATORS

There exist a variety of problems for digital forensic investigators in the field of password cracking. One major problem is the speed of password cracking. Due to improvements in security, the amount of time it takes to crack passwords encrypted with the latest secure algorithms is increasing. Thus, as security improves and advances, the number of challenges for the digital forensic investigator also increases. Another problem in the field of password cracking is the ambiguity in estimating whether or not the correct password will be found. If the password is very difficult and the encryption algorithm in use is secure, it is not possible to assume that a password will definitely be cracked. Also, a measure of probability for determining the chances of finding a correct password does not exist, since the password being searched for is unknown.

As just discussed, the two major problems in the field of password cracking are the decreasing speed of password cracking and the ambiguity of the outcome of the password cracking process. These two major problems are the cause of even more problems and challenges for the digital forensic investigator. Password cracking is slow for strong passwords and passwords that are encrypted with a strong algorithm. Due to such slow speeds, it takes a lot of time to crack those passwords. This in turn leads to an increase in the amount of time for the associated legal cases to be resolved. The slow speed of password cracking and thus legal resolution also causes an escalation in the costs involved. Also, the cost involved for resolving the legal case becomes very high due to the large amount of time required. It is therefore a challenge to manage the time and also costs for password recovery.

There are also other challenges for the digital forensic investigator. If the information to be accessed is in encrypted form, it is difficult to know whether or not the information is relevant to the investigation. Therefore, it is challenging to identify relevant information without cracking the password. Depending on the case, the digital forensic investigator may also have other challenges with regards to accessing the evidence. There may be social engineering challenges—for example, social engineering the suspect to allow an investigator to stealthily steal the password or gain clues to get the password. There may also be technical challenges—for example, there may be technical challenges for the forensic investigator to gain access to the suspect's computer over the Internet.

In the field of password cracking, there exists a variety of advice for the digital forensic investigator. As a result, there are no standard best practices available to the forensic investigator.

1.2 MOTIVATION

This research has been motivated by the other research reports published in the area of password cracking. Previous research, which will be outlined in this section, has been done mostly in the areas of password security and digital forensics. All the past studies reviewed below are motivated towards improving password cracking speed and efficiency. Therefore, the research performed helps digital forensic investigators to reduce password cracking costs.

The researchers Thing and Ying (2009) researched methods to improve speed and efficiency for password cracking. Rainbow tables were previously the most efficient method of cracking passwords. In their research they proposed a method that utilises 50% of the storage space required by rainbow tables. The success rate achieved with the new methodology is the same as that of rainbow tables (See section 2.2.4 for an explanation of rainbow tables). Therefore the research results by Thing and Ying provide useful information for digital forensic investigators to improve password cracking speed and efficiency. Thus, the research results are also helpful for forensic investigators seeking to reduce password cracking costs.

Graves (2008) researched and implemented rainbow tables on the nVidia graphics card. Graves called this implementation 'IseCrack'. By the use of IseCrack, Graves achieved very fast password cracking speeds and also higher success rates. Graves' research is thus useful for digital forensic investigators seeking to improve password cracking speed and success rates with the use of graphics cards. Forensic investigators using his implementation may also gain cost benefits for password cracking cases.

The researchers Weir, Aggarwal, Medeiros, and Glodek (2009) discussed a new approach for password cracking. This approach consisted of the use of Probabilistic Context-Free Grammar (PCFG) based on a training set of previously disclosed passwords. The researchers demonstrated that their new approach improves the password cracking process in comparison to traditional password cracking methods. Therefore, the research results presented by Weir et al. (2009) can provide useful information to digital forensic investigators. Forensic investigators may use their approach to improve the password-recovery process. Since the process is improved, the forensic investigator may gain cost advantages over password cracking by the use of regular methods.

Dandass (2008) implemented an FPGA-based hardware implementation of the PKCS#5 technique published by the RSA. The PKCS#5 technique is used for generating password-derived encryption keys. Dandass showed that this hardware implementation can be used for improving password cracking performance. Therefore, the study done by Dandass provides useful information to forensic investigators. Forensic investigators may use the implementation by Dandass and gain improved performance. Thus, in this manner, the forensic investigator may also be able to reduce password cracking costs.

Oechslin (2003) proposed an improved way to perform cryptanalysis based on improving the cryptanalytic time–memory trade-off method. He also found a better way to calculate the pre-calculated data that is required for the purpose of cryptanalysis. Thus, as result of his work, Oechslin also increased the password cracking speed of Microsoft Windows hashes. In this manner, the research performed by Oechslin can help the forensic investigator gain improved performance over password cracking speeds for Microsoft Windows hashes. Thus, in this area, the forensic investigator can gain cost advantages for relevant password cracking cases.

For more studies performed in the area of password cracking, see Chapter 2 and Chapter 3 section 3.1. All the studies described above are motivated towards improving password cracking speed and performance. Therefore, the past studies are beneficial to the forensic investigator as they provide cost benefits. However, all of the aforementioned research is aimed towards increasing speed and performance by researching and improving the inherent technology in use. None of the past studies relate to the procedures involved in planning and estimating the cost of a password cracking case. The reason no such work has been performed may be due to the inherent impossibility of identifying the outcome of the password cracking process. There is also ambiguity involved in judging the amount of time it takes to crack a password.

Therefore, there is a motivation to perform research in order to plan and gain control over the costs or time required for password cracking. Hence, this research proposes the use of a budgeting model to plan and allocate resources for password cracking cases.

1.3 EXPECTED FINDINGS

The research to be performed will be based on a budgeting model (described in section 3.2.1). The experiment will be a simulation of hypothetical password cracking cases and will be planned in advance using the budgeting model and an experimental design. The time budgets for groups or blocks of passwords will be calculated and the resources needed for password cracking will be allocated. Then, the budgeting model and the experimental design will be followed and evaluated. The data to be collected will consist of the password cracking times. Finally, the budgeting model will be evaluated on the criterion of whether all the passwords were cracked within the allocated time budget.

This research expects all of the passwords to be cracked within the allocated time budget, since the budgeting model will be based on the relevant theory within the area of password cracking. It also expects the exact password cracking times for all the passwords to be variable in nature. It is expected that this research will be able to add to the knowledge of best-practice advice for forensic investigators. Also, the evaluation of the research design and entire research process will help in understanding any improvements to the budgeting model or the experimental processes. Since the experimental processes will be a simulation of hypothetical cases, the evaluation may also help to further add to best-practice knowledge and advice for digital forensic investigators. Digital forensic investigators will thus be able to use the knowledge or advice and apply it to real life cases (within constraints and limitations).

1.4 CONCLUSION AND STRUCTURE OF THESIS

This chapter has described the background of the study of passwords. Advantages and disadvantages of the use of passwords have been discussed; the disadvantages include the use of a password by a malicious person to encrypt incriminating information. Due to the security provided by passwords and encryption technologies, there exist challenges for digital forensic investigators seeking to gain access to relevant encrypted evidence.

The challenges faced by the forensic investigator in the field of password cracking were discussed in section 1.1. These challenges included the decreasing speed of password cracking and the ambiguity of the outcome of the password cracking process. These two main problems are the source of other problems, including slow investigation time, an increase in the costs of investigations, and an increase in the time taken for legal cases.

The factors motivating this research were discussed in section 1.2, where the past studies motivating this research were reviewed. Those published reports have motivated this effort to study and evaluate the processes involved in planning and gaining control over the costs and time required for password cracking. The current research proposes the use of a budgeting model to plan and allocate resources for password cracking cases. The expected outcomes of the research were discussed in section 1.3.

This thesis has been structured as follows. This chapter consists of an introduction to the topic of research.

Chapter 2 consists of the literature review, which defines the theory required for the research study.

Chapter 3 specifies the methodology of this research study. First it reviews the studies similar to this research, which help to derive a methodology for this research. Then it defines the proposed budgeting model and experimental design, the research design, and the research questions and hypotheses.

Chapter 4 then reports the research findings. The procedures followed in the research and deviations that were encountered are outlined. Then, the research findings are reported along with their analysis and presentation.

The research questions are answered in Chapter 5 and the research hypotheses are tested. The research design is evaluated and the budgeting procedure recommended for forensic investigators in Chapter 5.

Chapter 6 consists of a summary of findings and recommendations for further research. The references used and appendices then follow.

Chapter 2

Literature Review

2.0 INTRODUCTION

Digital forensic investigators often come across password-protected documents or data during the investigation of digital evidence. Overcoming the password protection to gain access to evidentiary data is a desirable objective for all digital forensic investigators. One of the most common ways of overcoming password protection is password cracking (Casey, 2004). There are various techniques and tools that are used by digital forensic investigators to crack passwords. The most popular tool is the Password Recovery Toolkit (PRTK) by AccessData (Casey, 2004). As stated by Marechal (2008), the most up-to-date techniques to crack passwords involve the use of brute-force cracking techniques. These techniques can be further improved using Markov chains and rainbow tables. The brute-force technique guesses the password, hashes it, and then compares it with the original hash of the password (Rowan, 2009). If all the possible password combinations are tried for this attack, this process will eventually crack the password. This may be one of the main reasons why the brute force technique is widely used.

This chapter presents a literature review of password cracking software and techniques in order to provide a scope for the art and to identify problems and issues associated with it. In section 2.1, different password schemes are discussed to define some of the options available to an investigator. In section 2.2, a range of password cracking techniques are elaborated, and in section 2.3, the tools available are defined. Section 2.4 introduces the concept of cost. Any password may be cracked with infinite resources; however, an investigator has resources that are limited by the scope of any particular inquiry and therefore requires an estimate of the time costs involved. Section 2.5 is a tabulation of fourteen related research papers published on the topic of password cracking. Section 2.6 presents a summary of issues and problems and section 2.7 brings a conclusion to the chapter.

2.1 PASSWORDS AND PASSWORD RECOVERY OPTIONS

Passwords, along with their corresponding usernames, are used as a method of authentication. Authentication means verification of the fact that the user wishing to gain access is who he or she claims to be (Raval & Fichadia, 2007). During forensic investigations, a variety of items might be password protected. These items could include operating systems, CMOS, various files or documents (for example Microsoft Word documents and even compressed files), and even software. Each of these different items has a different password scheme. Operating system password schemes, particularly the password scheme of the Unix operating system, are discussed below.

It is important for forensic investigators to be able to overcome the password authentication process and gain access to the relevant item or evidence of interest. There are many methods to overcome password protection or to recover passwords. Some of these methods are explained below.

2.1.1 Password Schemes in Operating Systems

Many popular operating systems support the use of passwords for the purpose of authentication. However, each operating system utilises a different mechanism for its implementation (Raval & Fichadia, 2007).

In order for a person to gain access to the operating system, he or she must enter his or her credentials. These credentials consist of a username and a password. The username is used to identify the user that is logging in. The password, which is usually kept secret and known only to the user, is used as a means to verify whether the person is who he or she claims to be. After the username and password are entered and submitted, the operating system checks to see if they are correct. If the credentials entered are correct, the operating system grants access to the user. However, if the credentials entered are incorrect, the operating system denies access.

For the operating system to perform these checks these credentials must be stored in memory, since the operating system needs a stored value to compare with the entered value. If the passwords were stored in clear text, there would be major security issues, including any user being able to read all the available credentials of other users. Therefore, operating systems store each password in an encrypted form known as the password hash. The password hash is of a fixed length and is nonsensical in nature. It is derived from the original password by passing it through a one-way hash function (Raval & Fichadia, 2007).

A one-way hash function can derive the password hash by using the password as a value for the function. However, after the hash is calculated, it is impossible for the function to be reversed to derive the original password from the password hash. Also, the one-way hash function is designed to be 'collision-free', such that different passwords generate different hash values (Raval & Fichadia, 2007). In order for authentication to take place, the user enters his or her password, which is passed through the one-way hash function. If the output generated matches the stored hash value, then the user is allowed access. This mechanism is demonstrated in Figure 2.1.

Some operating systems generate the password hash by passing the password along with an additional value called a 'salt' through the hash function to increase randomness. Thus, a 'salt' is a short, random string of characters that is appended to the password to increase security (Salomon, 2006). One of the most common values used as a salt is the username (Raval & Fichadia, 2007).

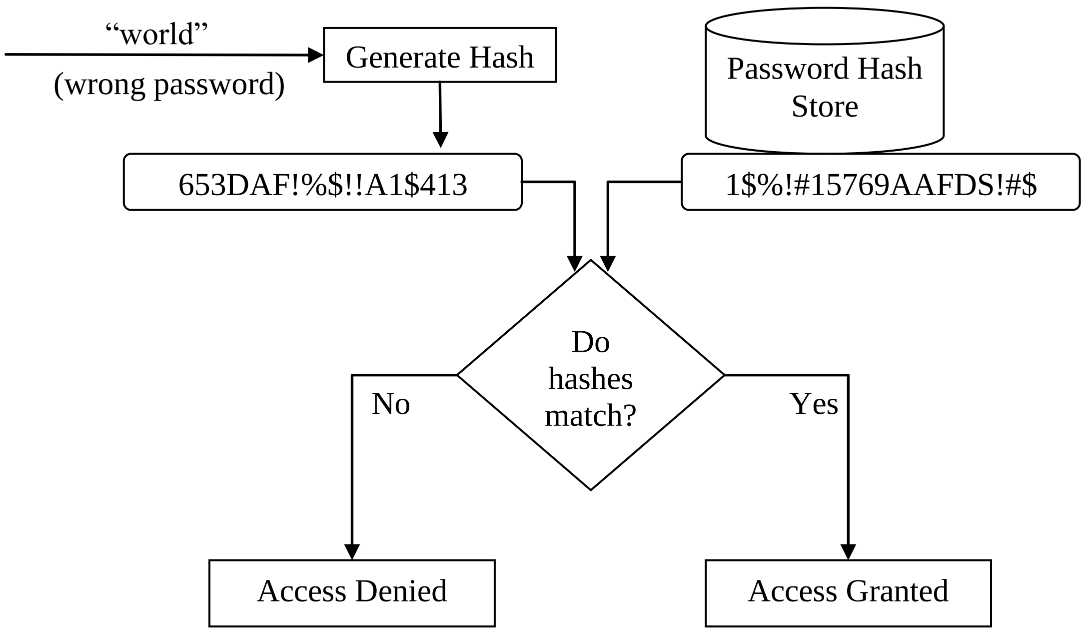

Figure 2.1: Password hashing and verification (Raval & Fichadia, 2007, p. 180)

2.1.2 Unix/Linux MD5 Password Scheme

Unix/Linux supports a variety of password hashing schemes. The classical Unix operating system uses the DES algorithm for password hashing, while newer versions of Unix/Linux support the MD5 hashing algorithm. If the MD5 algorithm is used, theoretically Unix/Linux could support passwords of unlimited length. Some flavours of Unix support passwords with a maximum length of 256 characters (Peikari & Chuvakin, 2004).

MD5 is much more computationally intensive and hence much more secure than the DES algorithm (Peikari & Chuvakin, 2004). MD5 takes approximately 20 times the amount of CPU processing time for hashing compared to the DES algorithm (Toxen, 2003). As discussed in section 2.1.1, the MD5 algorithm generally adds a salt value to the password while hashing it. For the MD5 algorithm in Linux, there are 2^{128} different possibilities for the salt value (Toxen, 2003).

Unix/Linux operating systems use the hash routine function called 'crypt()' for the purpose of encrypting a password (Stallings, 2006). The crypt() function uses the Modular Crypt Format or MCF encoding for the purpose of encrypting passwords in MD5 or DES (Viega & Messier, 2003). After generating the password hash using the crypt() function, Linux then stores the password hash in the /etc/shadow or the /etc/passwd file (Peikari & Chuvakin, 2004). Thus, a password encrypted in Linux by using Modular Crypt Format could be shown as:

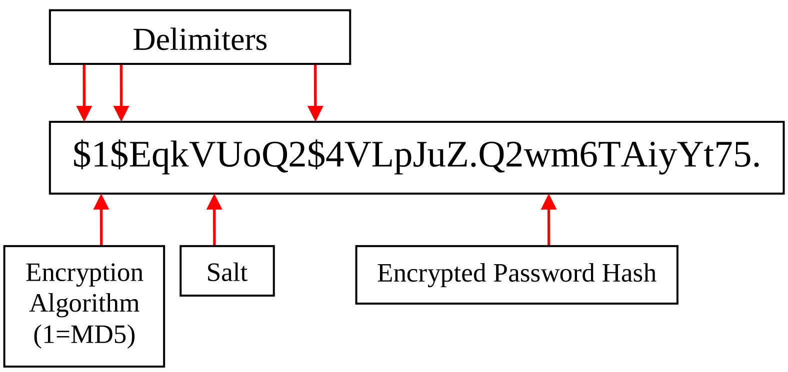

Figure 2.2: Example MD5 password hash value in Modular Crypt Format (MCF)

As shown in figure 2.2, the MD5 password hash value in Modular Crypt Format consists of three fields. The '$' sign is the delimiter for separating the three values in the fields. The various fields in Modular Crypt Format and their purposes are shown in table 2.1.

Table 2.1: The Modular Crypt Format (Garfinkel, Spafford, & Schwartz, 2003, p. 88)

Field	**Purpose**	**Notes**
#1	Specifies encryption algorithm	1 specifies MD5, 2 specifies Blowfish.
#2	Salt	Limited to 16 characters.
#3	Encrypted password hash	Hash value without salt.

2.1.3 Password Recovery Options

In forensic investigations, investigators may come across many items of interest which may be password protected. Therefore, investigators need to overcome password protection to gain information of evidentiary value. Many types of items may be password-protected, and many different methods can be used for obtaining the password. The method to be used depends on the circumstance. For example, if a law enforcement agency has online access to the criminal's computer, they may make use of a key logger to obtain the relevant password. Likewise, if the law enforcement agency has seized the computer, the password can be cracked by using various password cracking techniques. Thus, different password-protected items require different methods for password recovery. Some of the types of password-protected items along with their potential methods for recovery are discussed in this section.

The various options for password recovery are as follows:

i) Jumping/pulling the battery: Computers may contain CMOS or BIOS passwords at startup. If so, recovering the CMOS password is necessary in order to boot the computer. Many methods may be used to bypass the CMOS password lock or recover the password. These methods include, but are not limited to: removing the CMOS battery, manipulating the BIOS jumper settings, short-circuiting the chip, interrogating the suspect, or trying out the default BIOS passwords (Britz, 2009, p. 331).

ii) Cracking: Some documents, software and compressed files may contain passwords. These passwords also serve the purpose of encrypting the information within such files. In order to recover such passwords, password cracking software such as Password Recovery Toolkit or John the Ripper may be used. These software programs make use of password cracking techniques such as dictionary attacks and brute force attacks (Britz, 2009, p. 336).

iii) Brute force/social engineering: In order to crack the password, the investigator may choose to make a profile of the suspect. This profile could include information such as favourite colour, place of birth, pet's name, partner's name, and other personal details. The password cracking can then take place either manually or by using a dictionary of the suspect's profile (Britz, 2009, p. 337).

iv) Key loggers: A key logger is any software or hardware that can record every key pressed on the computer (Burnett & Kleiman, 2006). Investigators may use key loggers to wiretap the suspect's computer without his or her knowledge (provided they are legally permitted to do so, for example, by the issuing of a warrant), thereby gaining all important information including passwords.

v) Keyboard acoustics: Research done by Berger, Wool, and Yeredor (2006) provides the basis for password recovery using the sound produced by the keyboard. Using this technique, it is possible to recover passwords whose lengths are between seven and thirteen characters from a recording of the clicks made by the keyboard. The researchers demonstrated a success rate of 90% while recovering words of ten or more characters. Thus, it could be possible for a forensic investigator to recover passwords using sound recordings of the clicks made while typing a password.

vi) Page files: Lee, Savoldi, Lee, and Lim (2007) have demonstrated the possibility of recovering passwords and sensitive information by collecting the page file. This can be done by using a page file collection tool that is also suitable for live forensics. Thus, a forensic investigator may perform live forensics to recover page file data in order to retrieve passwords and other sensitive information in order to solve the case.

vii) Program defaults: Many programs store their passwords at default locations, sometimes even in clear text. Investigators may find it helpful to compile a list of standard defaults for password location. They might also find it helpful to compile a library of program-specific crackers (Britz, 2009, p. 337).

viii) Analysis of hard disk: It may be possible to identify passwords during intensive analysis of the hard disk. Some forensic tools provide features for sniffing the hard disk for usernames and passwords. Passwords may also be located by analysing other sources such as slack space and swap files (Britz, 2009, p. 337).

ix) Same password: Once one password is obtained, the investigator may try to use the same password in other places. This method relies on the fact that it is a common human tendency to use the same password in various places (Britz, 2009, p. 337).

All these options may be used by investigators for the purpose of password recovery.

2.1.4 Existing Password Recovery Strategies using PRTK/DNA

AccessData's Password Recovery Toolkit (PRTK) and Distributed Network Attack (DNA) software suites are reviewed below in section 2.2.3. In order to retrieve passwords, there are certain strategies that are recommended by AccessData Corp (2006). According to AccessData Corp (2006), password owners usually choose a password that can be easily remembered. Thus, the password is in a language known to the owner. The password is usually related to some aspect of the owner's life. It is also possible that new passwords may be derived from previous passwords used by the owner.

Summarised from the whitepaper published by AccessData Corp (2006), the strategy recommended to forensic investigators for recovering passwords with the use of PRTK and DNA is as follows:

i) Find out the languages known to the owner. Also, determine the codepage or keyboard settings of the owner's computer.

ii) Search the owner's location for clues such as handwritten notes or passwords. If possible, create a customised dictionary based on the owner's biographical information.

iii) If possible, create a custom dictionary from the owner's hard drive images.

iv) Set levels of password processing in PRTK or DNA as required.

The more information that is available about the owner of the password, the more likely it is that the investigator will be able to crack his/her password (AccessData Corp, 2006).

Therefore:

> Password recovery is waiting for the set of target passwords to be tried against the encrypted file. Once all the background information about the creator of the password has been gathered and submitted to the recovery process, time becomes the limiting factor to recovering a password. A machine's speed, or the amount of machines available, will have a noticeable effect on the password recovery. (AccessData Corp, 2006, p. 3)

2.2 PASSWORD CRACKING TECHNIQUES

One of the most common methods of overcoming password protection is password cracking (Casey, 2004). As discussed earlier, passwords are often stored in an encrypted form. Many techniques can be used to recover the plain-text password from the encrypted form. Some of the most popular techniques are password decryption, brute-force attacks, dictionary attacks, and rainbow tables. These password cracking techniques are discussed in detail in this section.

2.2.1 Password Decryption

This method targets the weakness of the hashing algorithm used to encrypt and store the password. If the algorithm used for encrypting the password is not strong enough or is implemented incorrectly, then it is possible to crack any password, regardless of how good the password may be.

One method of password decryption is called *one-byte padding*. In this method, one byte of the program is changed, allowing the password to be decrypted (Shinder & Tittel, 2002). Another technique of password decryption is called the *known plain-text method*. In this method, the cracker has obtained some plain-text versions along with the encrypted versions of the files (Shinder & Tittel, 2002). This allows the cracker to decrypt other files encrypted using the same algorithm.

This method may be useful for investigators when passwords are encrypted using weak hashing algorithms.

2.2.2 Brute force

This method of password cracking is used when the encryption algorithm used to encrypt the password is strong and the password cannot be decrypted. In such a case, passwords are cracked by guessing. Many tools can be used to guess the password. Every guess is encrypted by using the same algorithm as the original password, and the hash value is compared to the hash value of the original password. If the two match then the guess made is correct and the password has been recovered (Skoudis, 2007).

In the brute-force password cracking technique, every possible permutation and combination of the password is tried until the correct password is found (Dube & Gulati, 2005). Therefore, such an attack will eventually succeed in guessing the right password. However, this attack is only suitable for short passwords. For longer passwords, the attack is very time-consuming, since the sample space of every permutation and combination is extremely large. Thus, bigger passwords can take many years to recover using this password cracking technique. Due to these reasons, this form of attack is used as a last resort by investigators.

2.2.3 Dictionary attacks

This method of password cracking is also used when the encryption algorithm is strong and the password cannot be decrypted. Therefore, in this case as well, the password is cracked by guessing. The guess is encrypted and then the output is compared with the given hash value. If the two match, then the password is recovered (Skoudis, 2007). In the dictionary attack, the values to be guessed are taken from the dictionary. Thus, all the words in the dictionary are tried until the correct match is found (Dube & Gulati, 2005). The dictionary may also be modified by social engineering or by adding the user’s personal information. The personal information could include things such as favourite colour, partner’s name, date of birth, place of birth, and any other important information that the user might use to set their password. Many popular tools such as Password Recovery Toolkit, John the Ripper, and even L0phtCrack support this password cracking technique.

Since all possible permutations and combinations of passwords are not tried, there are chances of this attack being unsuccessful. However, this is a popular password cracking technique since many users set their passwords as a common word—either from the dictionary or from something personal (Britz, 2009).

2.2.4 Rainbow Tables

The password cracking process works by guessing passwords. Every guess is encrypted by using the same hashing algorithm used to hash the password. Then the output hash produced by the guessed password and the hash of the original password are compared. If the two are the same, then the guessed password is the required password (Skoudis, 2003).

Rainbow tables are used to enhance this process and improve speed. This improvement of speed is achieved by the principle of time–memory trade-off (Oechslin, 2003). In rainbow tables, hashes for billions of passwords are pre-computed and stored. The time required to generate these tables is large, but once generated, they can be re-used (Burnett & Kleiman, 2006). With the use of rainbow tables, passwords can be cracked a lot more quickly. This is because it is faster to look up the hash value from the table than to compute it each time to compare it with the hash value of the original password.

There is a variety of software that supports rainbow tables. This includes but is not limited to AccessData's Password Recovery Toolkit (PRTK), AccessData's Distributed Network Attack (DNA), and L0phtcrack 6 (AccessData Corp., 2010; Potter, 2009). Thus, rainbow tables are an improvement over brute-force and dictionary attacks.

2.3 LITERATURE REVIEW OF TOOLS USED FOR PASSWORD CRACKING

There are many tools available that can be used for the purpose of password cracking. Covering all the tools available is beyond the scope of this literature review. Therefore, this section contains a literature review of some of the most popular password cracking tools. These tools are John the Ripper, L0phtCrack, and AccessData's Password Recovery Toolkit (Casey, 2004; Wiles & Reyes, 2007; McClure, Scambray & Kurtz, 2009).

2.3.1 John the Ripper

John the Ripper is one of the oldest and most popular tools for cracking passwords (Wiles & Reyes, 2007). There are two versions of the tool available: the free, open-source version, and the pro version. This password cracking tool is command-line based and does not have a graphical user interface. Since the tool is open-source, there are many upgrades and patches available out of the box. These patches provide a range of additional features and add-ons to crack many different encryption algorithms. There are various cracking modes available. These modes consist of wordlist mode, single crack mode, incremental mode, and external mode (Openwall Project, 2010). John the Ripper also contains a built-in compiler, which can be used for programming your own cracking mode. John the Ripper can deliver speeds of an average of 800,000 passwords per second. It also occupies very minimal CPU resources (Cisneros, Bliss & Garcia, 2006).

2.3.2 L0phtCrack

L0phtCrack is a tool developed and released by L0pht Holdings, LLC. The L0phtCrack 6 tool is mainly used for auditing or security testing and for recovering passwords. There are four versions of L0phtCrack available for purchase: Professional, Administrator, Site and Consultant. Each of the four versions has a variety of features and is meant for a variety of audiences (L0pht Holdings, 2009).

L0phtCrack 6 can run well on 64-bit Windows and has a Windows Vista–style user interface. L0phtCrack 6 supports several types of attacks, namely dictionary crack, dictionary/brute hybrid crack, precomputed crack, and brute-force crack (Potter, 2009; L0pht Holdings, 2009). As summarised from the L0phtCrack official documentation, L0phtCrack 6 supports auditing for six different types of password hashes:

i) The LM Hash (for Windows)
ii) The NTLM Hash (for Windows)
iii) The LM Challenge Response
iv) The NTLM Challenge Response
v) Unix MD5-encoded password files
vi) Unix DES-encoded password files.

L0phtCrack 6 supports the retrieval of password hashes from the Windows operating system with admin privileges. L0phtCrack 6 can also retrieve password hashes remotely from a domain controller. Another useful feature supported by L0phtCrack 6 is the ability to crack password hashes obtained by packet-sniffing the network with the use of WinPcap (L0pht Holdings, 2009).

A major advancement in the field of password cracking, as discussed above in section 2.2.4, is the use of precomputed hashes or rainbow tables. L0phtCrack6 has the ability to use rainbow tables, allowing for quicker password recovery. The HashGen utility is also bundled up with the L0phtCrack 6 application. The HashGen utility allows users to create their own custom rainbow tables by specifying various parameters such as size and complexity of passwords to precompute (Potter, 2009). An example screenshot of L0phtCrack 6 performing a Unix password audit is shown in Figure 2.3.

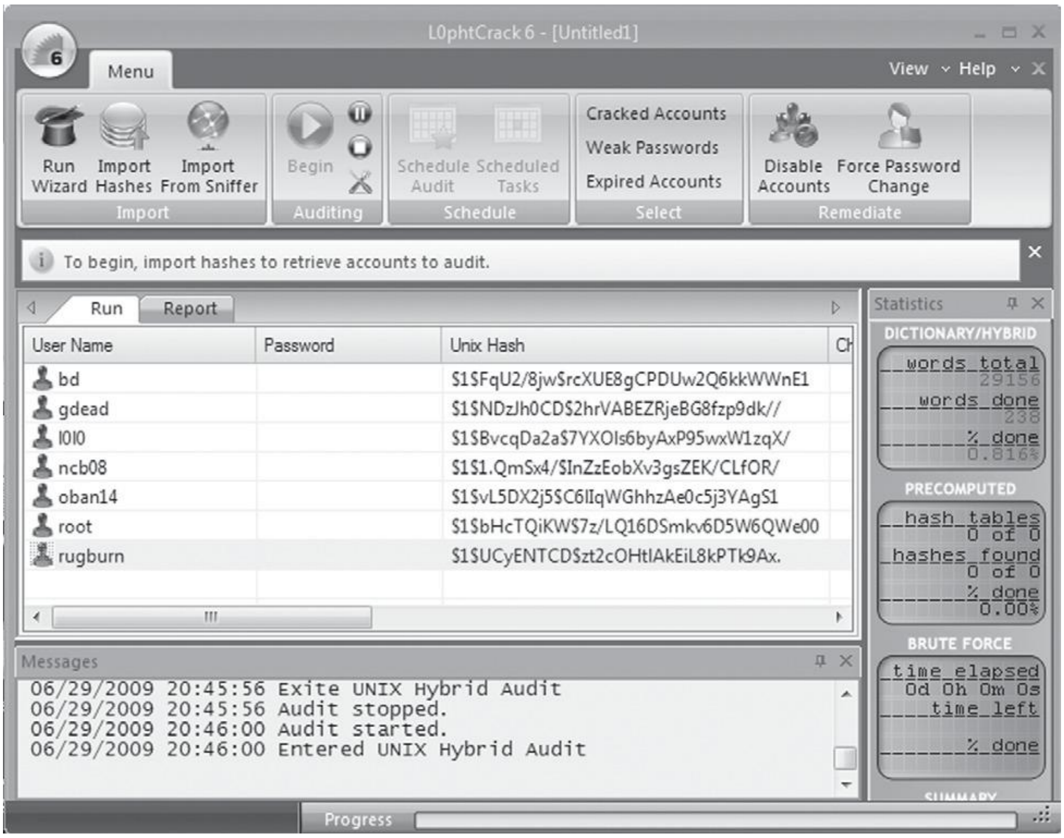

Figure 2.3: UNIX password audit (Potter, 2009, p.17)

Graphical reporting is another useful feature supported by L0phtCrack 6. The contents of L0phtCrack 6's reporting tab consist of results based on risk, character sets used, method of attack, how compromised the passwords are, and also the length of passwords (Potter, 2009). An example of L0phtCrack 6's report is shown in Figure 2.4.

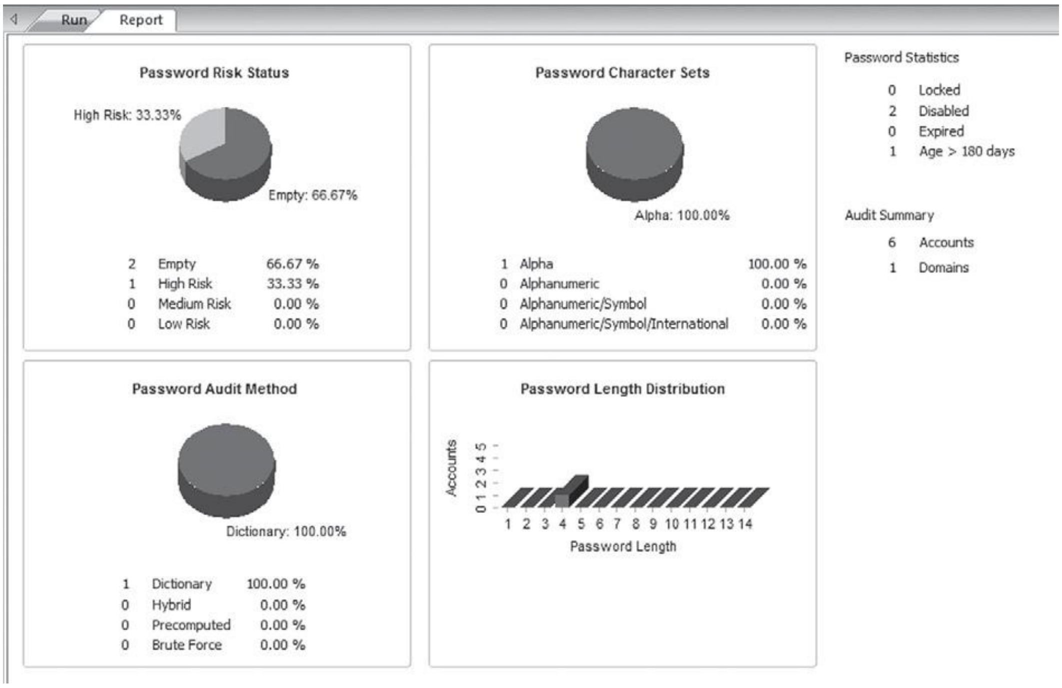

Figure 2.4: An example of reporting in L0phtCrack 6 (Potter, 2009, p.17)

2.3.3 Password Recovery Toolkit & Distributed Network Attack

Password Recovery Toolkit (PRTK) and Distributed Network Attack (DNA) are commercially available password-recovery software packages released by AccessData Corp. The target audience for PRTK and DNA are law enforcement agencies, corporate security, and IT professionals. PRTK is also targeted at individual users such as administrators who wish to recover lost passwords. PRTK and DNA are the most powerful and versatile password-recovery programs available on the market. PRTK supports a wide variety of file types and encryption standards; DNA has the same features as PRTK, but supports harnessing the power of several computers for cracking passwords (Casey, 2004; AccessData Corp., 2010).

PRTK and DNA support a wide variety of password cracking attacks. Adapted from the Distributed Network Attack and Password Recovery Toolkit user guide by AccessData Corp. (2010), the various attacks these packages support are:

i) Decryption attack: As discussed in section 2.2.1, the decryption attack decrypts the password that locks the file.

ii) Dictionary attack: As discussed in section 2.2.3, the dictionary attack uses words from the dictionary and also applies variations (or rules) to these words to crack the password.

iii) Keyspace attack: In the keyspace attack, each and every possible combination of keys are tried to decrypt the file.

iv) Reset: With the reset feature, it is possible to reset the key used for opening the file to a custom key, thus eliminating the need for recovering the key.

PRTK and DNA have a wide variety of features. Adapted from the Distributed Network Attack and Password Recovery Toolkit user guide by AccessData Corp. (2010), some of the important features are:

i) Hash files: The hash files feature calculates a unique hash value for the file whose password is to be recovered. PRTK and DNA calculate the hash value of the file when it is added for processing. After the password recovery process is over, the hash value is calculated again. If the two hash values calculated before and after the password recovery process match, then it implies that the file's contents have remained unchanged during the whole process. Thus, the file-hashing feature is useful for law enforcement personnel to maintain the integrity of evidence (AccessData Corp., 2010, p. 2).

ii) Recover multi-language passwords: Multiple language dictionaries are bundled along with PRTK and DNA. Thus, it is possible to recover multi-language passwords (AccessData Corp., 2010, p. 2).

iii) Generate reports: PRTK and DNA can print password-recovery job reports in pdf format (AccessData Corp., 2010, p. 2).

iv) Open encrypted files: PRTK and DNA support opening encrypted files using recovered passwords. However, the application used for opening the files must be installed on the same computer. Recovered files can also be copied or moved to another location (AccessData Corp., 2010, p. 3).

v) Customise rules and dictionaries: PRTK and DNA allow users to create or import custom dictionaries. It is also possible to create biographical dictionaries that allow you to enter the suspect's biographical information to aid in password recovery. PRTK and DNA also maintain a golden dictionary that contains previously recovered passwords. They also support creation and customisation of password recovery rules, or modifications that can be made to dictionary keywords to be used for guessing passwords. For example, it is possible to add any prefixes or postfixes and to choose any variations in character sets to customise password recovery.

vi) Use add-ons: PRTK and DNA support various add-on products. These add-ons include rainbow tables and Portable Office Rainbow Tables released by AccessData Corp. PRTK and DNA also support Tableau TACC1441 Hardware accelerator (TACC). The use of TACC reduces dictionary-based password recovery times. Thus, it is possible to increase speed and accuracy with the use of various add-ons.

2.4 COSTING PASSWORD CRACKING

There exist a variety of problems for digital forensic investigators when applying any particular password cracking technique. One of these problems is estimation of the time taken to complete the cracking. As a result, it is a problem to estimate the cost of such an investigation. This section describes costing for password cracking and also discusses the importance of estimating costs for the purpose of password cracking in forensic investigations.

2.4.2 Costing

Costing, or cost accounting, means the "establishment of budgets, standard costs and actual costs of operations, processes, activities or products; and the analysis of variances, profitability, or the social use of funds" (Lucey, 2002, p.1). This subsection defines full cost and the two types of full costs, namely direct costs and indirect costs.

2.4.2.1 Full Cost

Full cost refers to the total amount of resources, measured in monetary terms, that are utilised to achieve a particular objective (Atrill, McLaney, Harvey, & Jenner, 2006).

When units are not identical, costing is calculated on a job-by-job basis. Since every password to be cracked is different, each password would have to be cracked on a job-by-job basis. Therefore, the costs for cracking passwords should also be calculated on a job-by-job basis. In order to find out the cost of a particular unit of output (or job), all of the direct costs associated with producing that unit are assigned to it. Each of the units of output is then also charged with a fair share of indirect costs. This is known as job costing and is demonstrated in Figure 2.5 below (Atrill et al., 2006).

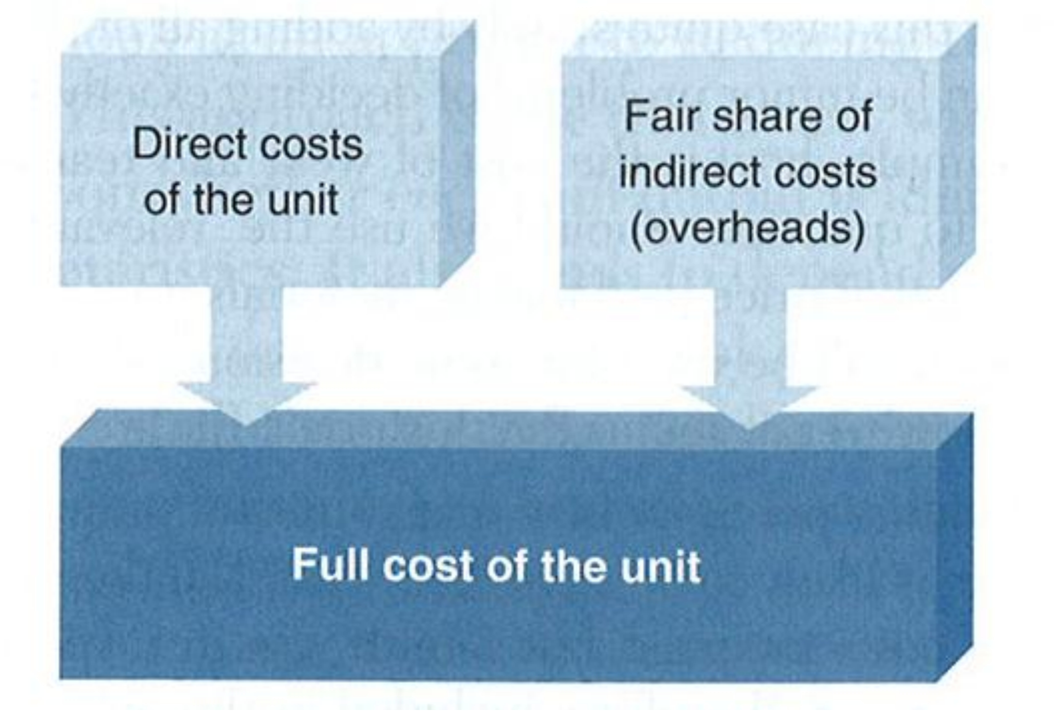

Figure 2.5: Full cost of a job (Atrill, McLaney, Harvey & Jenner, 2006, p. 332)

2.4.2.2 Costs for Cracking a Password

In the case of password cracking, the cost factors associated with cracking a password for each unit of output (i.e., each job) includes, but is not limited to the costs shown in Table 2.2.

Table 2.2: Example of costs associated with password cracking

Direct Costs	Time
Indirect Costs	Power; computer and equipment depreciation.

Thus, the cost of cracking a password can be calculated as:

Cost of cracking a password = Direct costs + Fair share of indirect costs.

The time required to crack a password determines the indirect costs such as power costs and computer and equipment depreciation costs. Thus, the amount of time spent would be directly related to other indirect costs. Because of this, if the amount of time is reduced or managed, it could be possible to reduce or manage the costs of cracking a password.

2.4.2.3 Time Estimation

As discussed above, time is the direct cost involved for the purpose of calculating the costs of cracking a password. It is difficult to estimate the time required to crack a password. If the password is 'len' characters in length and the character space contains 'alpha' number of characters, then the key space (the space containing all possible combinations of passwords) of the password 'k' would be $alpha^{len}$ (Rowan, 2009). Therefore, as the number of characters and the length of the password increases, its key space increases exponentially. This can be shown in Table 2.3 below.

Table 2.3: Comparison of character space, length, key space and maximum time taken to crack a password (Adapted from Rowan, 2009, p. 5)

Character Space	**Length**	**Key Space**	**Max. Time to Crack @ 500,000 Passwords/Second**
10	4	10000	1 sec
10	5	100000	1 sec
10	6	1000000	2 sec
26	4	456976	1 sec
26	5	11881376	24 sec
26	6	308915776	10 minutes
26	10	141167095653376	9 years

As shown in Table 2.3, a password whose length is 10 characters long, and which is chosen from a character key space of 26 characters, could potentially take up to a maximum of 9 years to crack. However, this does not give a good estimate of the amount of time needed to crack the password, since it may be cracked in the first guess, or millionth guess, or even the last guess. If it is cracked in the first guess, it would take 1 second to crack that password at 500,000 passwords per second cracking speed; if in the millionth guess, it would be cracked in 2 seconds. However, if it is guessed at the last guess, then it would be calculated in 9 years. Therefore, there exists a large ambiguity with regards to the accurate amount of time required to crack such a password.

2.4.2.4 Cost Estimation and its Importance

Cost estimation is very important to forensic investigators. As discussed above, time is the direct cost involved in cracking passwords. Also, it has been demonstrated in the previous subsection that the time required for cracking a password is uncertain. This makes it difficult to estimate the cost of the job before it is done.

People are generally very conscious of costs, even for the purpose of forensic investigations (Casey, 2006). Due to many problems in the realm of forensic investigations, including increasing backlogs and an increasing number of crime cases for investigation, forensic investigators outsource password cracking and other work in the private sector to the lowest bidder (Casey, 2006). This demonstrates the importance of cost to forensic investigations. Also, because of the importance of cost, there is the good chance investigators will cut corners and trade justice for low cost (Casey, 2006). Therefore, estimating the costs of password cracking in forensic investigations is an important issue.

2.5 KEY AREAS OF RESEARCH

Many researchers have performed research in the areas of password cracking. These researchers, along with their key areas of research, are presented in Table 2.4. The key factors from their research that influence and motivate this research have also been listed.

Table 2.4: Key areas of research

No.	**Researchers**	**Key Areas of Research**	**Usefulness to Forensic Investigators**
1)	Thing and Ying, 2009	Researched methods to improve speed and efficiency of password cracking. Proposed a method that utilises 50% of the storage space of rainbow tables, yet has the same success rate.	Useful in the field of password cracking, as it may help increase speed of cracking passwords. Hence, could also prove a cost benefit for forensic investigators.
2)	Graves, 2008	Researched and implemented IseCrack, which is an implementation of rainbow tables on the nVidia graphics card (GPU) and achieved a high speed and success rates.	Useful resource in terms of password cracking utilising the commonly acquirable nVidia graphics card; good for improving speed of password cracking. Hence, could also be useful as a cost benefit for forensic investigators.
3)	Dell'Amico, Michiardi, and Roudier, 2010	Measured the password strength for existing real life datasets of passwords from three different sources. They also compared the state-of-the-art password cracking techniques for this purpose, including Markov chains and dictionary attacks.	Useful resource in providing inspiring insights for the purpose of making better password-recovery tools. Studying user habits is helpful in improving password-recovery speeds and hence could also provide cost advantages.
4)	Frykholm and Juels, 2001	Proposed a fault-tolerant method, in which a high-entropy password is	Useful in studying an alternative password security implementation scheme, and

		generated by a low-entropy password. This scheme allows the users to recover passwords if they remember certain parts of the password correctly.	how to recover passwords from such a scheme. Also could provide useful password-recovery information for forensic investigators. Thus, could also be a cost benefit to forensic investigators.
5)	Weir, Aggarwal, Medeiros, and Glodek, 2009	Discussed a new method for password cracking by creating a probabilistic context-free grammar based upon a training set of previously disclosed passwords. Also discussed how the new scheme improves password cracking and recovery compared with the traditional methods of password cracking.	Useful for improving password cracking mechanisms and speed. Could provide a good cost benefit for forensic investigators.
6)	Dandass, 2008	Implemented a FPGA-based hardware implementation of the PKCS#5 technique published by the RSA laboratories for the purpose of generating password-derived encryption keys. Discussed how performance can be improved using a hardware-based password-	Useful for improving password cracking mechanisms and speed. Thus, use of such hardware platforms could provide a good cost benefit to forensic investigators.

		generating unit for the purpose of performing a dictionary attack.	
7)	Oechslin, 2003	Proposed an improved way to perform cryptanalysis, by improving on the cryptanalytic time–memory trade-off method. Also succeeded in finding a better way to calculate the pre-calculated data needed during cryptanalysis, and in turn also increased the password cracking speed for Microsoft Windows Password Hashes.	Useful for improving password cracking mechanisms and speed. Thus, could provide a good cost benefit to forensic investigators.
8)	Mentens, Batina, Preneel, and Verbauwhede , 2005	Presented hardware architecture for Unix password cracking using Hellman's time–memory trade off.	Useful for improving password cracking mechanisms and speed. Use of such hardware platforms could provide a good cost benefit to forensic investigators.
9)	Clayton and Bond, 2003	Performed attacks against the IBM 4758, which is used in retail banking to protect ATM infrastructure. The author used FPGA-based hardware implementation to crack DES keys. This resulted in an increase in	Useful for improving password cracking mechanisms and speed. Also provides good insights into different systems and computers used in the industry for banking purposes. Such information may be useful for solving

		the speed of cracking and the attack was successful.	bank-related cases. Password cracking study and architecture in the paper could provide a good cost benefit to forensic investigators in the area of password cracking.
10)	Potter, 2009	Reviewed the latest L0phtcrack version 6 password cracking tool.	Useful resource in order to gain information on a popular password cracking tool. Useful for background study and also for comparing the different tools in order to give best-practice advice for forensic investigators.
11)	Berger, Wool, and Yeredor, 2006	Researched and presented a keyboard acoustic–based password cracking method. This cracks the password based on the signals provided by the click of the keyboard while typing the password.	Useful resource and a good method of cracking passwords based on keyboard acoustics. Studying different techniques of password cracking could prove useful to provide for best-practice advice for forensic investigators.
12)	Lee, Savoldi, Lee, and Lim, 2007	Researched password-recovery options with the use of page file evidence collection tools. Discussed how page files store critically sensitive information, including passwords. Also presented	Useful in the field of password cracking and forensics. Could also be useful for best-practice advice for forensic investigators.

		countermeasures.	
13)	Craiger, Swauger, and Marberry, 2005	Discussed the use of forensics tools and techniques for locating encrypted information and also information hidden by steganography. Also covered the tools and techniques for cracking passwords in digital forensics to recover obfuscated data.	Useful information for cracking passwords in forensic investigations. Paper provides good research in order to provide for best-practice advice for forensic investigators.
14)	Mookhey, 2004	Presented a list of open-source tools used for security and vulnerability-checking purposes, as well as their uses in forensics. Also reviewed password cracking and recovery tools.	Useful for selecting the right tools for password cracking, improving mechanisms, and providing comparisons with tools. Such information could be useful best-practice advice for forensic investigators.

As shown in Table 2.4 (numbers 1 to 9), many researchers such as Thing and Ying, Graves, Dell'Amico et al., Frykholm and Juels, Weir et al., Dandass, Oechslin, Mentens et al., and Clayton and Bond have performed research that could potentially help provide cost benefits to forensic investigators. Also presented in Table 2.4 (numbers 10 to 14), researchers such as Potter, Berger et al. , Lee et al., Craiger et al., and Mookhey have performed research in the field of password cracking to help provide best-practice advice to forensic investigators.

2.6 KEY PROBLEMS AND ISSUES SUMMARY

The above summary of previous research shows that many questions have been answered regarding methods, techniques and software tools. The principal problem being investigated has been not the actual cracking of a password (which is inevitable) but rather the optimisation of resources. The key areas and problems for forensic investigators in the field of password cracking are:

- Reducing the time it takes to crack a password in order to quickly process and finish legal cases.
- Reducing the time it takes to crack a password in order to reduce the cost of password cracking.
- Managing the costs for password recovery. That is, accurately estimating the costs required for password recovery and accurately allocating resources for password cracking.
- Dealing with challenges of password recovery for suspects who use strong encryption and strong passwords.
- Tracing evidence to correctly identify encrypted information. It is also difficult to identify whether or not encrypted information is relevant to investigation, unless the password is already cracked or information is decrypted.
- Other challenges include alternate means of gaining evidence protected by cryptographic information, for example, social engineering challenges (social engineering the suspect in order to stealthily steal his or her password), and technical challenges to gain access to suspect's computer over the internet.
- Due to the variety of advice, there is no standard best-practice advice for password recovery in forensic investigations.

2.7 CONCLUSION

Investigators are faced with an access problem to data that has been password-protected. The evidence may reside in systems, files, or elements of spreadsheets and databases. Sections 2.1 and 2.2 have shown that there are many options for an investigator to approach the access problem. However, the overriding concern is cost efficiency. In sections 2.3 and 2.4, the tools and costs have been elaborated upon. In section 2.5, a summary of relevant research has been presented, and in section 2.6 the principal problem of costing and budgeting has been specified. In the next chapter, a methodology is to be developed to research the problem of password-recovery costing and budgeting.

Chapter 3

Research Methodology

3.0 INTRODUCTION

In Chapter 2, the relevant literature on password cracking was reviewed in order to define the topic and the scope of the current research. A short tabulation of relevant published research in the area was also made. In this chapter, a selection of published works is reviewed in depth to locate the methodologies other researchers have used to do password cracking (Section 3.1). These reviews are used to guide the development of a research methodology (Section 3.2). From the key problem areas of cost identified in Section 2.6, the research question, sub-questions, and hypothesis are developed. The data requirements and analysis are specified in Section 3.3. Section 3.4 discusses the limitations of the proposed research methodology and Section 3.5 provides a conclusion.

3.1 REVIEW OF SIMILAR STUDIES

A review of five published studies is made in the following subsections in order to identify the methodologies used to do research in the area of password cracking. Each study has a different perspective on the problem, but when viewed together an approach, a methodology, and methods can be derived to construct the methodology of this research. The first paper looks at survey method, the second at cluster methods, the third cost reduction methods, the fourth probabilistic methods, and the fifth parallel clustering techniques.

3.1.1 A Survey of, and Improvements to, Password Security

The researcher Klein (1990) conducted a survey in which he outlined common threats to password security. He also performed password cracking tests to identify the existence of these threats. Consequently, he proposed the use of a proactive password checker for the purpose of improving password security.

Klein (1990) addressed the most common threat to password security, which, as identified in Section 2.1.2, is that a cracker may gain access to the Unix password file. The cracker may then attempt to crack the passwords in the password file using dictionary attacks and brute-force techniques. Klein conducted a survey amongst friends and acquaintances, in order to gain a sample set of passwords on which to perform the password cracking test. He also requested a copy of their password files for the purpose of performing these tests. Since it benefited his respondents as well, he also provided them with a vulnerability report for their systems, based on the results of his research. However, due to the sensitive nature of the information, the researcher received only a small fraction of replies and password files. From this small fraction, he chose one password file to represent the sample for his password cracking experiment. The password file contained about 13,797 account entries.

Using that password file as the sample, Klein (1990) utilised various techniques to crack the passwords, principally dictionary attacks. The password cracking guesses were performed by selecting a password guess along with a salt. Klein grouped together passwords with common salt values to speed up the testing. The methods of attack included trying the user's personal details, common dictionary words, foreign dictionary words, and also various permutations of all of these words. All inter-dictionary and intra-dictionary duplicates were eliminated to reduce the search space. The researcher used four computers, each of which was capable of guessing about 750 passwords per second. The tests were carried out for 12 CPU months.

As a result of these tests, Klein (1990) cracked 24.2% of all of the passwords from the sample. These results have been tabulated by including the dictionary type used, size of dictionary, duplicates eliminated, search size, number of matches, percentage of total, and cost/benefit ratio. The cost/benefit ratio was calculated by dividing the number of matches by the search size. The username/account name search yielded the highest cost/benefit ratio, whereas the names of asteroids yielded the lowest cost/benefit ratio. It is also worth noting that the largest number of passwords that were cracked were six characters in length. Due to these results, Klein proposed the use of a proactive password checker consisting of all of the rules that were used for cracking, which would not allow the user to select a vulnerable password in the first place. Hence, Klein recommended the pro-active password checker to cover up the security vulnerability in order to make the crackable 24.2% of passwords secure.

3.1.2 An Analysis and Comparison of Clustered Password Crackers

Frichot (2004) conducted research and analysis of clustered password cracking software John the Ripper. The main intention of his research was to compare the two software packages John the Ripper and Cisilia, which utilise the Open Mosix and Beowulf styles of parallel computing respectively. However, Frichot was unable to perform a comparison of the two, since he encountered problems using Cisilia. Therefore, he conducted an analysis and highlighted issues in regards to clustered password cracking with the use of John the Ripper.

Frichot's (2004) background study on passwords and password cracking methods is similar to the ones identified in the literature review in Chapter 2. In order to perform his research, Frichot made use of two clusters of computers. The first cluster consisted of 13 nodes and was set to the Beowulf configuration in order to test John the Ripper. The second cluster consisted of 14 nodes and was set to the OpenMosix configuration for testing Cisilia. Both of these clusters were set up on the Linux platform.

For the main experiment, Frichot (2004) created password samples in Microsoft's LAN manager (LANMAN) format. The sample passwords consisted of manually created passwords, which were created so as to cover a broad spectrum of password quality. The researcher used Williams' (2001) algorithm to determine password quality and created passwords that had quality ratings from 5 to 14 (cited in Frichot, 2004). Frichot performed password cracking tests on both the Beowulf cluster and the OpenMosix cluster. The data he collected for the Beowulf cluster included half the cipher-text password, half the plain-text password, username, a digit representing which half of the password was cracked, and the amount of time it took to crack the password. It was essential to note which half of the password was cracked, since the LANMAN hashing algorithm supports a maximum of seven-character hashing. Thus, for passwords longer than seven characters, LANMAN divides them into separate blocks of seven characters and then hashes them individually. The resulting LANMAN hash for such a password would be all of these individual hashes appended together. The data collected for the OpenMosix cluster included the time taken to crack the password, the username, and the plaintext (i.e. cracked) password. Frichot performed two tests for each of the clusters to measure the reliability of the tools. To ensure comparability of the two clustered password crackers and to complete the work on time, the researcher performed the experiment for a maximum of three days.

After the experiment was performed, Frichot (2004) collected the results. He found Cisilia's results to be inconsistent and unstable; thus, he discarded Cisilia's results and was unable to compare them with John the Ripper's. John the Ripper provided more consistent results. Hence, Frichot instead analysed the results for John the Ripper and highlighted issues with regards to clustered password cracking by the use of John the Ripper. According to his results, John the Ripper cracked 34 password hashes in the first 24 hours. In the next two days, John the Ripper cracked four more password hashes, resulting in a total of 38 cracked passwords in three days.

According to the analysis of the results, 12 passwords were completely cracked (i.e., both separate LANMAN hashes for the passwords were completely cracked). Also, 27 passwords were half-cracked with only the first seven characters of the password cracked. The remainder of 11 passwords were not cracked at all. Frichot (2004) compared the relative strength of the password and the time it took to crack. The password strength vs. time comparison showed no relation as most of the easy and hard passwords were cracked within the first few hours of the first day itself. Two medium-strength passwords were cracked on the second day. The reason given by the researcher for such results was that LANMAN supported seven characters for hashing. Thus, it was easy to crack two separate hashes. Frichot also observed that the passwords that took longer than a day to crack contained special characters in them. He further observed that there were probable issues with regards to inconsistent truncating and padding of passwords either by the LANMAN algorithm or by John the Ripper.

Based on the analysis, Frichot (2004) suggested future research. His suggestions include further research in the topic of clustered password cracking using stronger encryption algorithms such as MD5. He also suggested further analysis into a universal metric for password strength and further research with regards to hardware clusters.

3.1.3 Password Cracking using Sony Playstations

The researchers Kleinhans, Butts, and Shenoi (2009) evaluated the benefits of using the Sony Playstation 3 (PS3) for the purpose of password cracking. The PS3 was used for research since it offered benefits of higher computing power at lower costs. The researchers also suggested using multiple PS3s for parallel processing to increase computational power for password cracking. They also described a distributed framework meant for law enforcement agencies, which would be useful to crack passwords in an efficient and cost-effective manner.

Kleinhans et al. (2009) presented a literature review of the cell broadband engine architecture. This architecture was designed for the purpose of gaming; however, Kleinhans et al. identified its use for other high-performance computing applications. The researchers conducted a series of password cracking experiments, the main goal of which was to evaluate the PS3 as a viable and cost-effective option when compared with the current options of Intel and AMD processors. For the experiments, the researchers wrote a C/C++ code for encrypting passwords using the MD5 encryption algorithm. The experiments consisted of two stages of password generation. The first phase consisted of the generation of all possible password strings from a set of 72 characters (26 uppercase, 26 lowercase, 10 numbers. and 10 special characters). The second phase consisted of generating the MD5 password hashes by the use of the above-mentioned C/C++ program. The researchers also verified the functionality of the code by comparing the result of the program with the Linux MD5 generator. For the experiments, the password generation stages were allocated evenly amongst all the system processors. The experiments were conducted using a strict brute-force implementation for password cracking. The various experiments were conducted by generating passwords of lengths of four, five, six and eight characters.

For the purpose of evaluating the results of the experiments, the researchers used three metrics, namely passwords per second, computational time, and cost efficiency. The number of passwords per second was the mean of the number of passwords generated in the given time intervals. The computational time was defined as the estimated time required for generating the entire password space. The cost efficiency metric was defined as the number of passwords per second divided by the dollar cost of the computing platform.

Kleinhans et al. (2009) compared the passwords per second of the various lengths of passwords generated for each processor present in AMD, Intel and the PS3 systems. For the password length of four, one AMD processor generated roughly 800K passwords per second, one Intel processor generated roughly 350K passwords per second, and one PS3 processor generated roughly 500K passwords per second. Considering the entire system, AMD generated nearly 3.2 million passwords per second, Intel generated 1.4 million passwords per second, and the PS3 system generated 3.1 million passwords per second. For all the four lengths of four, five, six and eight characters, the PS3 performed within the 4% range of the AMD system and significantly outperformed the Intel system. Considering the computational time metric, the PS3 and AMD had comparable computational time whereas the Intel system had approximately 47% higher computational time. The cost efficiency results for the PS3, AMD and Intel were 7,700 passwords per second per dollar, 2,100 passwords per second per dollar, and 750 passwords per second per dollar, respectively.

Thus, Kleinhans et al. (2009) came to the conclusion that the PS3 is more cost-efficient than AMD or Intel. They also concluded, based on the performance results, that the PS3 is a viable option for password cracking in law enforcement agencies. Based on the conclusions, Kleinhans et al. also described and recommended the use of a distributed password cracking framework. For future work, they would refine and optimise the parallel usage of PS3 architecture. The researchers would also implement the distributed password cracking framework.

3.1.4 Password Strength: an Empirical Analysis

The researchers Dell'Amico, Michiardi and Roudier (2010) conducted a study to compare and evaluate the effectiveness of various password cracking attacks using known datasets of passwords. The empirical analysis study was conducted to answer the research question: "given a number of guesses, what is the probability that a state-of-the-art attacker will be able to break a password?"(Dell'Amico et al., 2010, p.1). Based on the study, the researchers found the 'diminishing returns' principle to hold true. They found that weak passwords are cracked easily; however, as the attack goes on, the probability of finding the correct passwords decreases. Dell'Amico et al. propose that the results of the study would help to evaluate the security of passwords and serve as a basis for developing effective pro-active password checkers and security auditing tools.

Dell'Amico et al. (2010) discuss the importance of evaluating the resilience of passwords to guessing attacks. The resilience can be measured by comparing the number of guesses (i.e., the search space size) against the percentage of passwords successfully cracked. The attack model used would determine the cost of each guess for the attacker. Combining the cost of each guess with the size of the search space would result in a cost-benefit analysis for guessing-based attacks on password authentication systems. Dell'Amico et al. also conducted a literature review of the previous work done in the area. They compared search space size versus number of cracked passwords using various attack methods, including dictionary attacks, brute-force attacks, dictionary mangling, probabilistic context-free grammars, and Markov chains. They conducted the experiments on three large datasets of passwords, which differed in terms of application, domain, and user localisation.

The three datasets of passwords obtained by the researchers were the 'Italian dataset', 'Finnish dataset' and 'MySpace dataset'. The Italian dataset contained unencrypted passwords from an Italian instant messaging server running on the XMPP protocol. The Finnish dataset contained a publicly released password set in encrypted and unencrypted format from different Finnish forum websites. The researchers considered the unencrypted passwords from this dataset for the purpose of their study. The MySpace dataset contained disclosed passwords that were obtained via phishing attacks on the MySpace website. The MySpace dataset was the largest dataset of passwords. By observing the three datasets, Dell'Amico et al. (2010) found that there were many cases of the same passwords. Perhaps the reason for finding the same passwords may be coincidence or the same users registering under different usernames and setting the same password. The average length of the passwords in the three datasets was about eight characters. Dell'Amico et al. also observed that in all the three datasets, 10% of the passwords consisted of non-alphanumeric characters. In the MySpace dataset, 20% of the users complied with the password policy of inserting a non-alphabetic character by appending a 1 to the end of the password. The researchers also observed that some of the users of the Italian datasets chose very strong passwords with hard-to-detect structures.

Dell'Amico et al. (2010) conducted a simulation of dictionary attacks on the password datasets. The dictionaries used were the paid ones available from the John the Ripper website, which contain basic words from 21 different languages. The dictionary attacks were conducted only on the Italian and the Finnish datasets. MySpace's password policy required non-alphabetical characters to be a part of the password. Thus, if a dictionary attack consisting of basic language keywords were to be conducted on the MySpace dataset, it would return no results due to the alphanumeric password requirement of MySpace. Hence, the attack was not conducted on the MySpace dataset.

The researchers displayed the results of the number of passwords matched from the dictionaries in a tabulated format. The results contained the various dictionaries in increasing order of size. The researchers also calculated the 'guess probability' for each dictionary. The 'guess probability' was the ratio of matched passwords from each dictionary divided by the size of the dictionary. The dictionaries had non-empty intersections, i.e., some of the Italian and Finnish words were in common. Thus the researchers found some common passwords in both the Finnish and Italian datasets. Another finding was that some common English words were also used as passwords, since most users probably spoke English as their second language. The most important finding from the dictionary attacks simulation was the principle of 'diminishing returns'. The probability of guessing a password sharply declined as the size of the dictionary increased. The researchers also found that the mnemonics dictionary was ineffective, since probably a very small number of people use mnemonics for their passwords.

The next experiment involved using dictionary-mangling rules. Dell'Amico et al. (2010) used John the Ripper to generate an extended dictionary by applying varying mangling rules to the biggest John the Ripper dictionary named 'all dictionaries'. The extended, mangled dictionary generated consisted of a search space size of approximately 148 million passwords. Another hand-tuned dictionary already consisting of mangled passwords was also used by the researchers. The hand-tuned mangled dictionary consisted of a search space of approximately 41 million passwords. Apart from using the mentioned two dictionaries, Dell'Amico et al. also made use of Probabilistic Context-Free Grammars (PCFG) to create dictionaries. For the purpose of PCFG, they randomly chose half of the passwords from each of the three datasets to create the PCFG training set. The PCFG training set was then applied to John the Ripper's 'all languages' dictionary to create three dictionaries for the three languages of English, Italian, and Finnish. The various search space sizes of the dictionaries generated for each of the three languages were approximately 1.45 million, 41 million, and 148 million, respectively. As a result of the tests, Dell'Amico et al. found the principle of diminishing returns to hold true for the mangling and PCFG attack tests as well.

3.1.5 Parallel Password Cracker: A Feasibility Study of Using Linux Clustering Techniques in Computer Forensics

The researcher Bengtsson (2007) conducted a feasibility study for using parallel clustering techniques for the purpose of computer forensics. For the purpose of conducting the feasibility study, he built a Linux-based high-performance computing cluster. In this cluster, Bengtsson developed a parallelised password cracking program using the Message Passing Interface (MPI). The purpose of conducting the research was to gain competence in using a high-performance computing cluster for applications like computer forensics, where high-performance computing is required.

Bengtsson (2007) outlined a literature review of the related work in the fields of high-performance computing clusters, Message Passing Interface, password complexity, password cracking methods and passwords in Linux. He asserted that only a few password cracking programs available are capable of running in parallel computing clusters. Some of these password cracking programs are open-source tools such as MPICracker that have modified source codes available. These modified source codes allow the tool to run in a computing clustered environment. Bengtsson also mentioned that John the Ripper has open source codes available to take advantage of parallel computing by the use of the Message Passing Interface. He went on to identify work done in the area such as openMosix being ported with John the Ripper and Cisilia. Bengtsson overviewed the high-performance clusters with the use of Linux hosted on computers with low-cost, off-the-shelf components. Also discussed in the overview was the importance of using a message-passing technique such as the Message Passing Interface to allow the nodes in the high-performance cluster to communicate.

Bengtsson (2007) also described password complexity issues similar to the ones described in Section 2.4.2.3. Various password cracking methods such as dictionary-based, brute-force, and rainbow tables were presented in the literature review. The password cracking methods were similar to the ones described in Section 2.2. Along with the password cracking techniques, Bengtsson also discussed passwords on Linux, which are similar to the ones described in Section 2.1.3.

Bengtsson (2007) performed the feasibility study by developing a high-performance computing cluster using low-cost, off-the-shelf components, on which he installed a suitable Linux distribution. He also developed a demo application on the Linux clusters with the aim of it being useful in the field of computer forensics. The cluster Bengtsson assembled comprised 42U 19-inch rack-mounted Beowulf High-Performance Computing cluster. The clusters contained six Iwill DK8ES motherboards and dual AMD Opteron 244 processors. Each CPU had dedicated 2GB of PC3200 REG ECC DIMM and also Dual Maxtor MaXLine III SATA-150 disks striped in a RAID 0 array. The CPUs also contained 3Com’s 3C996-SX 1000BASE-SX Network card and were connected to a cheap GSM712F Netgear fiber gigabit switch. The researcher installed the Slackware distribution of Linux along with MPI-1.2 implementation MPICH 1.2.6. For the purpose of execution, Bengtsson used the remote shell (rsh) service.

Bengtsson (2007) created a demo application for the purpose of testing. The application was developed in ANSI-C and was called ‘brutest’. The brutest application utilised the MPI library for parallelisation. A few bash scripts were also created for the purpose of testing. The brute-force attack was carried out on a root account stored in the Linux shadow file. A complex password was set so that the password cracking process could be carried out for a longer duration. Ten computers were used to carry out the password cracking experiment. Bengtsson tabulated the results displaying various factors including password length, MPI wall clock time(s), and estimated hashes per second. Using a character set of 26 characters from a to z, for the passwords of length 4, 5, 6, and 7, the estimated hashes generated per second were 18,781, 12,499, 12,478, and 12,383, respectively. Using a character set of 36 characters (i.e., characters a-z and numbers 1-10), for the passwords of length 4, 5, and 6, the estimated hashes per second were 17,302, 12,587, and 12,618, respectively. Based on these results, Bengtsson assumed an average speed of 12,500 hashes per second.

Based on the experiment, Bengtsson discussed the feasibility of using clustered computing, asserting that the cluster that was set up in his experiment could be used for high-performance computing applications such as forensics. Setting up a cluster by using off-the-shelf computer components and the Linux operating system makes it an affordable choice. The risks of using off-the-shelf components, however, include problems in the supply chain of parts required, technical difficulties assembling them, compatibility issues with hardware, and other potential problems as well. Bengtsson mentioned that performance optimisation is also possible. However, there are also various issues with optimising performance. The issues include, but are not limited to, scalability issues, increasing latency, caching problems, and memory leakage. Bengtsson also discussed the weakness of the MD5 algorithm used in the experiment. Hash collision is a possible risk for algorithms such as MD5. However, Bengtsson stated that the risk of hash collision was not a big problem for most applications whose passwords need to be cracked. He stated that shadow files were used for the purpose of convenience. The main intention of the study was to learn how to use parallelising code in a clustered environment.

Bengtsson (2007) concluded that building a cheap computing cluster for the purpose of parallel computing is a competitive alternative to traditional options. Parallel clusters can be used for solving problems such as password cracking and hence can speed up forensic investigations. Thus, labs, institutes, or even companies with small funds are capable of setting up a high-performance computing cluster using Message Passing Interface. Bengtsson recommended future work in various areas of forensics such as password cracking, data mining, database querying, statistics, pre-calculation of lookup tables, parallel rendering, and parallel signal processing.

3.2 RESEARCH DESIGN

The relevant literature has been reviewed in Chapter 2 to identify a key problem in the research area. The research question, sub-question, and hypotheses are derived from this literature review in Section 3.2.4. The review of similar studies above in Section 3.1 identifies methodologies others have used to do related research. In this section, the research design is specified by deriving a budgeting model from the Chapter 2 review and a design model from the five similar studies in Section 3.1.

3.2.1 Budgeting Model

The main goal of establishing a research design is to reduce the cost of password cracking. As discussed in Section 2.4.2.2, the cost of cracking passwords is directly related to the time it takes to crack the password. Thus, this research proposes the use of a budgeting methodology to achieve control over the time it takes to crack a password. Section 2.1.4 discussed the existing best practices for the purpose of password recovery. These best practices are meant for forensic investigators using the software tools Password Recovery Toolkit and Distributed Network Attack. Thus, a budgeting model has been described within this section after considering the existing best-practice strategies for password cracking. The best password cracking tools available in the market, as discussed in Section 2.3, are also considered for proposing the budgeting model.

For the purpose of the budgeting model, certain terms need to be defined. The terms are ‘case’, ‘population set’, ‘password cracking speed’, and ‘total time to crack the password’. These terms are described in the subsections to follow.

3.2.1.1 Case

A case is simply a password cracking job that needs to be done. Usually, in the field of forensics, it will be based on the crime case on hand. As every crime case is unique, it can be expected that the password cracking job for each case would also be unique. Thus, every password that needs to be cracked would have some relation to the suspect whose password needs to be cracked.

3.2.1.2 Population Set

A population set is a unique population of information within which the correct password may be found. As per the best practices discussed in Chapter 2, the information within which the correct password could be found would in some manner be related to the suspect or the owner of the password. Thus, while planning to crack the password, as per the budgeting model, it is essential to map the information related to the suspect and the password. The information may be in the form of particular language character sets, particular language dictionaries, hand-written notes, data on a hard drive, the suspect’s biographical information, or any other clues. All such information could be considered as populations. It is essential to convert the population of information such as data on hard drives or hand-written notes into dictionary files. In doing so, it would be possible to feed these dictionaries to the password cracking program. Let the number of words from the total population to be tested in order to find the correct password for each password cracking case be represented by ‘P’.

3.2.1.3 Password Cracking Speed

The password cracking speed is the average number of passwords tried per second. As mentioned in Section 2.1.4, several factors such as the machine’s speed and the number of machines available affect the amount of time required to crack passwords. Also, various factors such as the encryption strength decide the amount of time required for computing and testing. Thus, based on such factors and based on results from practical tests, it would be possible to determine the average speed taken to crack a password. Let the average password cracking speed be called ‘S’.

3.2.1.4 Total Time to Crack Password

As the name suggests, this is the total expected time for the password to be cracked. Let the total time required to crack the password be represented by ‘T’. After determining the two factors mentioned above (‘P’ and ‘S’), the total time to test the entire population of passwords to the encrypted file would thus be:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

Thus, $T = P / S$.

Thus, as per the budgeting model, each case would have a unique population set. Also, depending on the factors mentioned above in Section 3.2.1.3, each case would also have unique password cracking speed. Hence, total time to crack the password would also be unique to each case.

As per the proposed budgeting model, the investigator assumes the two known factors of Population ‘P’ and Speed ‘S’ to determine the total resources of time and processing power to allocate.

3.2.2 Experimental Design

Based on the budgeting model described in Section 3.2.1, an experimental design was developed; it is described in this section. The proposed experimental design consisted of many hypothetical password cracking ‘cases’ or jobs (as described in Section 3.2.1.1).

Each of the hypothetical cases consisted of a unique population set. The password to be cracked for each case was randomly chosen from the unique population set for the respective case. The unique population sets to be used for each of the hypothetical cases are discussed in Section 3.3.1. Thus the experiment was essentially a simulation of many password cracking jobs or cases.

Klein (1990) used four computers, Frichot (2004) used two clusters consisting of 13 and 14 computers, and Bengtsson (2007) used ten computers. Based on the available resources, this experiment consisted of a cluster of eight computers. The eight computers were connected in a VLAN and ran Windows XP. The software used for the password cracking experiment was AccessData's Distributed Network Attack. One computer served as the Distributed Network Attack Supervisor and the remaining seven computers served as the workers. The jobs needed to be added to the Distributed Network Attack supervisor, which allocated the password cracking workload to the seven computers. Distributed Network Attack supports the design and allocation of unique profiles for each job. Thus, each password cracking job had a unique profile associated with it. The unique profile consisted of the population set relevant to the password case.

The password sampling methodology and budget allocation methodology for the password samples are described in Sections 3.3.1 and Sections 3.3.2 respectively. Once the password samples were determined, the next phase was to create the accounts on Ubuntu Linux. Frichot (2004) created password samples in Microsoft's LAN manager (LANMAN) format. Kleinhans, Butts and Shenoi (2009) and Bengtsson (2007) created passwords using MD5 encryption. Similarly, this experiment consisted of MD5 encryption being used in Ubuntu Linux. Once the accounts were created, the /etc/shadow file was exported. The different account entries along with the password hash for each of the accounts were split and separated. Each file was then loaded on to Distributed Network Attack for cracking and each file was associated with its respective profile.

Klein (1990) carried out tests for 12 months and Frichot (2004) carried out his experiment for three days. Considering practical limitations of time and available resources, this experiment was run for seven days. As discussed in Section 3.3.2, the experiment was monitored regularly and the data was collected and evaluated in order to answer the research questions.

3.2.3 The Research Questions and Hypotheses

As discussed in Section 2.6, there are various key problems and issues in the field of password cracking in forensic investigations. Time is the main factor to consider for reducing the cost of password cracking (see Section 2.4.2.2). Thus, control over the time taken to crack a password could provide control over the cost. Therefore, based on the key problems and issues, this research aims to study and investigate the time issues with regards to password cracking.

Therefore, based on the key problems and issues presented in Section 2.6, the main research question was formulated as:

> *Q: What are the time implications of cracking passwords using the budgeting model?*

The sub-questions for the main question are as follows:

> *Q1: How many passwords can be cracked within the allocated time budget?*

> *Q2: How much time is required to crack all of the passwords in each of the given blocks?*

> *Q3: What are the guidelines for best-practice advice for digital forensic investigators in the field of cracking passwords by the use of AccessData Distributed Network Attack (DNA)?*

The hypotheses for the secondary research question Q1 are:

H0: All of the passwords can be cracked within the time budget allocated by the use of the budgeting model.

H1: None of the passwords can be cracked within the time budget allocated by the use of the budgeting model.

H2: Some but not all of the passwords can be cracked within the time budget allocated by the use of the budgeting model.

The research consisted of four phases:

- Phase 1 - Sampling (or Data Creation): As discussed in Section 3.3.1, the sampling phase consisted of the creation of password samples.

- Phase 2 - Time Budgeting (or Data Processing): As discussed in Section 3.3.2, the time budgeting phase consisted of calculating the time budget for the password samples.

- Phase 3 - Testing (or Data Collection): As discussed in Section 3.3.3, the main experiment consisted of the data collection or the testing phase.

- Phase 4 - Report Generation and Analysis (or Data Analysis): As discussed in Section 3.3.4, the data collected in phase 3 was analysed in order to answer the research questions.

The four phases of this research are shown in figure 3.1.

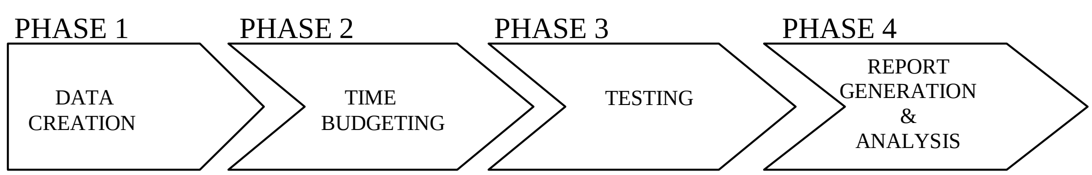

Figure 3.1: Four phases of research

3.3 DATA REQUIREMENTS

In order to test the research design developed in the previous section, the data requirements have been defined in this section. The first phase of the research consisted of creating sample passwords. A random password was selected from the various populations in consideration. The sampling phase was then followed by the main experiment, which is the next phase of the research design. After the main experiment, the next phase was to collect and analyse the results.

Section 3.3.1 will describe the sampling methodology in detail. The data processing method will be discussed in Section 3.3.2. The methods used for collecting the data will be discussed in Section 3.3.3. After the data collection, Section 3.3.4 describes the methods used for analysing the data collected in order to obtain the results. The last Section, 3.3.5, consists of the data map, which maps the flow of the research design.

3.3.1 Sampling

To conduct a password cracking experiment, it is necessary to create sample passwords. As per the experimental design in Section 3.2.2, each hypothetical case scenario consisted of a unique population set. For the purpose of this experiment, the unique population sets were each of the password cracking rules provided by AccessData’s DNA. Thus, the different password rules were the unique population set for each of the cases. Also, as per the research design in Section 3.2, the various rules provided by AccessData’s DNA are based on the intensity of the search or the size of the population to be tested. For the purpose of sampling, a password was randomly selected from each of these populations/rules. In order to choose the random password, the relevant dictionary entries were to be exported to Microsoft Excel. A random word was then to be selected by the use of the RAND() function in Microsoft Excel. (In practice, the experiment had to deviate slightly from these specifications; see Section 4.1.1.) After the random word was chosen, the relevant rule was applied to form the password. Each of these passwords was then associated with a custom DNA password cracking profile, which consisted of the rule category to which the password belonged. The profile consisted of the unique population relevant to the particular case/password.

For the main test, the order in which the passwords were tested was in ascending order of population size. Therefore, theoretically, the blocks consisting of smaller population sizes were assumed to be completed earlier. It is best practice not to add more than 50 files for password cracking at one time (AccessData Corp, 2010).

This limit is recommended by AccessData Corp in order to maintain the performance of the software and the computer system. Accordingly, having more than 50 passwords could hamper system performance and the password cracking process. Thus, for the main experiment, a block consisting of a maximum of 50 passwords was tested at a time. The sample consisted of a total of 200 passwords divided into four blocks. Table 3.1 shows an example of one of the passwords from the Basic level population size of 2,270,800.

Table 3.1: Example of password and profile to be used in the sample for main experiment

Rule ID	Description	Population Size	Password
Bas-2-32	Dictionary primary followed by a two digits search ([EN-1] Common-en-c.adf)	2,270,800	sensor56

For the above example, the associated custom profile consisted of the AccessData Rule '(BAS-2-32) Dictionary primary followed by a two digits search ([EN-1] Common-en-c.adf)'.

3.3.2 Data Processing

As described in Section 3.3.1, the sample space consisted of 200 passwords divided into four blocks. A time budget for each of these blocks was prepared. The time budget was calculated using the formula described in Section 3.2.1.4:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

Thus, to calculate P, for each block the total of population sizes to be tested was calculated. For password cracking speed (S), a benchmark value of 12,500 passwords per second per computer would be considered. The benchmark value was obtained from the MD5 clustered password cracking study by Bengtsson (2007). Since a total of eight computers would be used for the main experiment, the total password cracking speed was considered

12,500 x 8 = 100,000 passwords/second

After calculating the time budget, the total time value was rounded off to the highest minute value. The password cracking process for each block was run for the total rounded-off time budget for each block. The main experiment was assigned a budget to run for a total of seven days, considering the practical limitations of time and available resources.

3.3.3 Data Collection Method

The main variable of interest, as per the research design in Section 3.2, is the time taken for the passwords to crack. It is also essential to answer the research question of 'How many passwords can be cracked within the allocated time budget?' The Linux MD5 encrypted passwords were tested in blocks of 50 (see Section 3.3.2). These passwords were added one at a time whilst associating a profile for each. After completing the file/job addition procedure, the password was kept for cracking for the budgeted time allocated for that block.

Monitoring was essential to ensure the experiment ran smoothly. Thus, the password cracking process was actively monitored a minimum of every 12 hours, or as often as practically possible, or at the end of the allocated time budget for the given block, whichever was earliest. For the ease of monitoring the password cracking experiment, a 'monitoring/action performed log' was recorded. Table 3.2 displays the format of the log.

Table 3.2: Format of monitoring/action-performed log

S.No	Machine Number	Date and Time	Status / Action Performed	Next Scheduled Monitoring Time

The next block of passwords was added upon realisation of the end of the cracking process for the current block, or at the end of the time budget, whichever was earliest. The process mentioned in this section was carried out for the entire duration of seven days. The data was collected in the form of the report generated by AccessData DNA. The report included the main variable of interest, that is, the time taken to crack the password (AccessData Corp, 2010). Thus, data were collected in the form of the AccessData DNA password cracking report and the monitoring/action-performed logs.

3.3.4 Data Analysis Method

As was just mentioned, the data were collected in the form of AccessData's DNA password cracking report. The main variable of interest for the purpose of analysis is the time required to crack the passwords. Thus, the required information of usernames and the times required for the password to crack were filtered out from the AccessData DNA password cracking report and entered in a spreadsheet in Microsoft Excel. After the filtering and categorisation of the relevant data, it was possible to further investigate and present the data in the relevant visual form. For example, the data may be presented in bar charts or pie charts based on the relevancy. After the analysis was complete, it was possible to gain the answers to the research questions.

3.3.5 Data Map

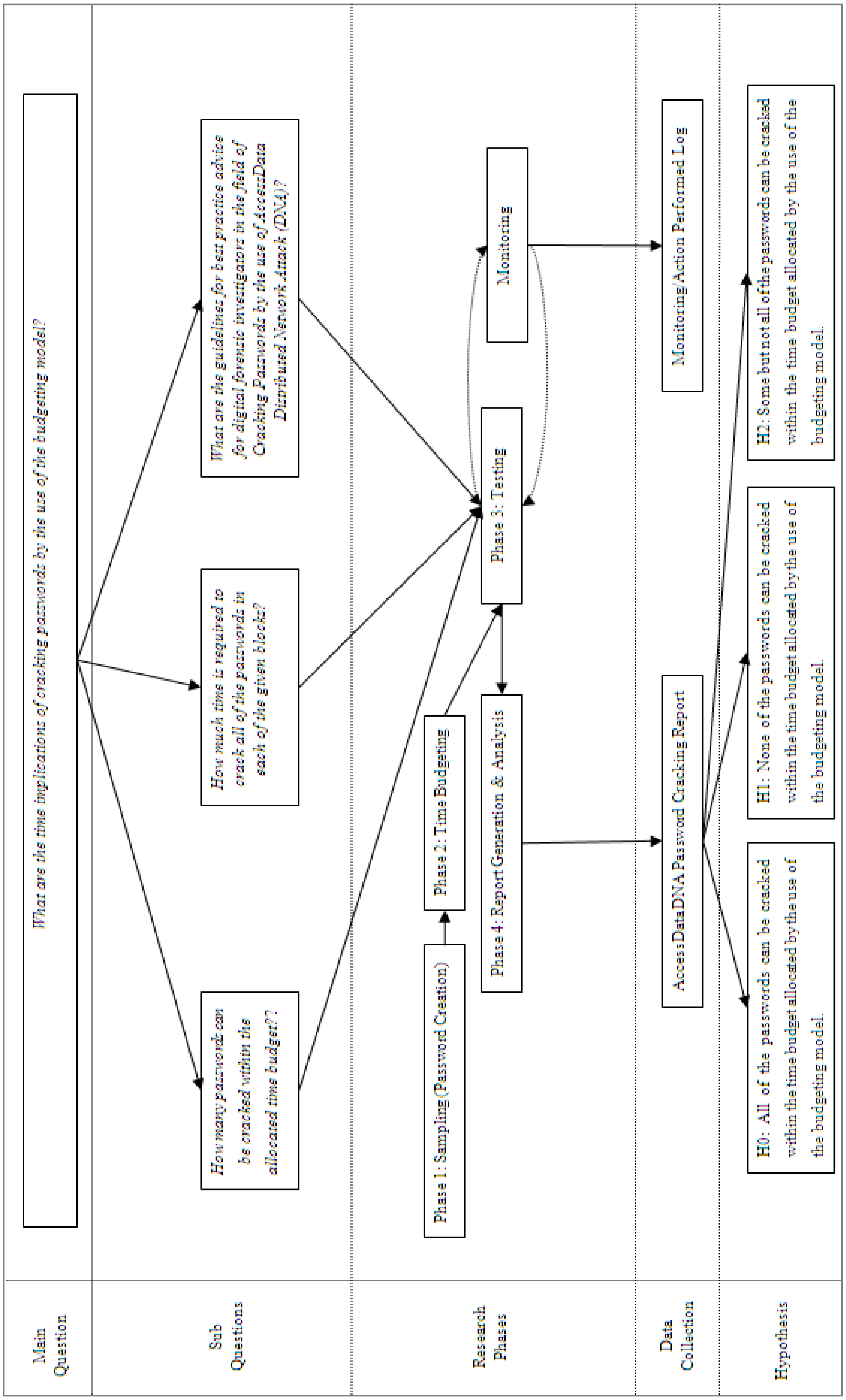

Figure 3.2: Research data map

3.4 LIMITATIONS OF THE RESEARCH

This research has selected one problem from many and then operationalised research around one relevant question that can add knowledge of the chosen problem. The proposed research can show what a budgeting method may look like, but this cannot be generalised to every context, case, or system. Also, in reality, every password cracking case is unique; therefore, there may be unknown complications when applying the budgeting method to real-life scenarios. A limitation of the experiment is that the test data were randomly generated. Thus, there was no way to verify whether the data set would be based on the characteristics of passwords one might find in real life. Also, the only way to assume the validity of the test data would be to attempt the password cracking procedure. Thus, the validity of the test data was unknown until after the experiment was complete. Consequently, another limitation of the research is that the forecasted outcomes based on the study may be used for general guidance within the declared constraints only.

3.5 CONCLUSION

This chapter has drawn together the theory section of the thesis. Chapter 2's findings in terms of relevant problems and issues have been used to derive a research question, sub-questions and hypotheses. The in-depth review of related studies in this chapter has guided the derivation of a working methodology to answer the research question. The experimentation can now proceed and the results are to be reported in Chapter 4.

Chapter 4

Research Findings

4.0 INTRODUCTION

The research experiment was carried out as per the methodology specified in Chapter 3. Some deviations from the specifications also occurred. All the were noted down and the effects they caused were taken into account. Hence, the experimental research processes were completed. The research consists of four phases: Sampling or Data Creation, Time Budgeting or Data Processing, Testing or Data Collection, and Report Generation and Analysis or Data Analysis. The results from the four phases of research carried out are reported in this chapter. The various findings from analysis and the presentation of the findings are also reported.

The chapter has been organised to first report the variations encountered in the experiment. Thus, the variations encountered in all the four phases of the experiment are described in Section 4.1. The test environment of the computer system for the main test and the data creation, processing, data collection, and report generation performed are reported in the fieldwork section, Section 4.2. The analyses performed on the reports generated are shown in Section 4.3. The data findings are presented in Section 4.4. The concluding remarks are found in Section 4.5.

4.1 VARIATIONS ENCOUNTERED IN EXPERIMENT

The experiment consisted of four phases (detailed in Section 3.2.3). The first phase was the data creation phase, in which the various sample passwords were created. The second phase was the time budgeting phase, in which the budgeted time to crack the passwords was calculated. The third phase was the testing phase, in which the password hashes were loaded onto Distributed Network Attack for the purpose of cracking. As discussed in Section 3.3.3, the experiment was monitored regularly to ensure its smooth operation. The fourth phase was the report generation and analysis phase. The variations encountered in all of the four phases are reported as below.

4.1.1 Data Creation

For the purpose of sampling, various hypothetical ‘cases’ were considered (as defined in section 3.2.2). Each of these cases had been assigned unique population sets from the English language. For each of these cases, a password was selected from the various population sets. The populations used were each of the rules provided by AccessData’s DNA. A total of 200 password rules were used for 200 cases. The rules presented in the software, however, were confusing. The dictionary files to be used for the rules were defined along with some of the general rules. However, with some of the rules, the dictionary files that were being used were not listed. After observing a common trend of dictionary files appearing in a certain order in the list for most of the rules, it was possible to determine which dictionary was being used with which rule. As per the common trend the order in which the dictionary files appeared in the rules were [EN-1] Common-en-c.adf, [EN-2] Miscellaneous-en-c.adf, [EN-3] Names-en-c.adf, [EN-4] General-1-en-c.adf and [EN-4] General-2-en-c.adf respectively. For example, for serial number 33, BAS-2-18 (please refer to rules list in the appendix), the dictionary for the rule ‘Dictionary primary reverse search’ is ‘[EN-1] Common-en-c.adf’. However, for serial number 34, BAS-2-18, the dictionary rule for ‘Dictionary primary reverse search’ is not mentioned. It was assumed to be [EN-2] Miscellaneous-en-c.adf, since the order in which dictionaries were included in the rules is mentioned above.

For the purpose of password creation, the dictionary entries were to be exported in Microsoft Excel. The random dictionary word was then to be chosen by the use of the RAND() function. However, it was not possible to export the list as the list could not be copied using the dictionary viewer in Distributed Network Attack. It was also not possible to export the dictionary files since the dictionary files were in AccessData’s .adf format. Thus, the dictionary words had to undergo manual random selection.

4.1.2 Time Budgeting

The time budgets for the various passwords were calculated in Microsoft Excel using the formula mentioned in the budgeting model in Section 3.2.1.4. No variations deviations from the research plan were encountered for calculating the time budgets of the various files.

4.1.3 Main Experiment

During the main experiment, the password hash files were added to Distributed Network Attack for the purpose of cracking. The main difficulty encountered was in adding the files to Distributed Network Attack. The various files had to be manually linked to a profile. Hence, each file had to be added one at a time. Thus, it was not possible to add the block of 50 files and link to them to the respective profile in one step. This resulted in different start times for each of the files. However, for the results, the start time and the end time both were considered in calculating the time taken to crack. Whilst monitoring the experiment, after the third block ended, the supervisor computer was unable to add any more jobs. Thus, the results were backed up and all the previously completed jobs were deleted from the list. The supervisor computer was then restarted and the final block of passwords was then added for cracking. The process steps were audited to assure the results were unaffected.

4.1.4 Report Generation and Analysis

After the experiment, the report generation feature of Distributed Network Attack was used to generate the report. The analysis of the results was carried out with the aid of Microsoft Excel. One unanticipated result was that some passwords returned no results. Thus, for some of the passwords, the entire password cracking process had completed without Distributed Network Attack being able to recover the password successfully. However, these passwords were still considered a part of the analysis. The reason for considering these passwords for analysis was that AccessData's Distributed Network Attack had run the entire population set to search for the correct passwords. Thus, in the context of this study, the results could be comparable to a password being found in the last attempt. However, the actual reason Distributed Network Attack returned no passwords for certain accounts was unknown and could not be investigated since it was outside the scope of this study. No other difficulties were encountered during this phase.

4.2 FIELDWORK

The specifications for the research design and data requirements were listed in Sections 3.2 and 3.3. The main experiment was performed along with certain deviations from the specifications listed in Chapter 3. The deviations encountered in the experiment were specified in Section 4.1.

The experimental fieldwork carried out is explained in this section. The test environment used for the experiment is discussed in Section 4.2.1. The sampling and time budgeting or data creation and processing are explained in Section 4.2.2. Finally, the testing or the data collection performed is explained in Section 4.2.3, along with a sample of the reports generated.

4.2.1 Test Environment

The main experiment environment consisted of eight computers connected in a Virtual Local Area Network. The eight computers had Windows XP loaded on them. Out of the eight computers, one was the supervisor computer in which Distributed Network Attack version 3.5.1 and version 1.6 of DNA's dongle drivers were loaded. The seven remaining computers were loaded with the worker modules available with Distributed Network Attack. The configuration of the eight computers is shown in Table 4.1.

Table 4.1: Configuration of computers in test environment

Operating System:	Windows XP PRO - Version 5.1.2600 Service Pack 2 Build 2600
Processor:	Intel (R) Core(TM)2 Duo CPU, E8400 @ 3.00 GHz
Physical Memory:	4096 MB with 1024 MB allocated to Intel (R) G41 Chipset on board display
HDD Capacity - Supervisor:	80 GB
HDD Capacity - Workers:	8.4 GB

Another computer with the same configuration and hard disk capacity of 8.4 GB was loaded with Ubuntu Linux 10.04 Long Term Support version. The computer with Ubuntu Linux was used to create the accounts whose passwords were to be cracked.

4.2.2 Data Creation and Processing

Before the main experiment could be performed, it was essential to create sample passwords for the various hypothetical cases. As defined in the sampling requirements, the hypothetical cases would consist of unique population sets, which in turn would consist of the rules provided by Distributed Network Attack. Distributed Network Attack supports several language dictionaries to which it would be possible to apply the rules. For the purpose of this experiment, the English language was used.

All of the rules in Distributed Network Attack were entered in a Microsoft Excel spreadsheet. The values consisted of a unique serial number, the rule ID, a description, and the population size. The description of the rule consisted of a description of the type of test along with the exact dictionary file that was being used. The values were then sorted by ascending order of population size and categorised in groups or blocks of 50, as per the previously defined sampling requirements. The first 200 entries were then considered, thus allowing for 4 blocks consisting of 50 entries each to be considered. After the values were entered and categorised, two more columns were created to enter the random password for each of the population sets and also to calculate the time budget required for each case.

For the purpose of creating random passwords, the Distributed Network Attack dictionary viewer was used to view the relevant dictionary files. A word was randomly selected from the relevant AccessData dictionary, the rules were manually applied to it, and the password value was written in the spreadsheet. For rules without any dictionaries, the password was chosen by the use of the relevant character sets table present in the AccessData Whitepapers – Character Sets.

The budgeted time required in seconds was then calculated using the formula:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

For example, for a population size of 22,708, the time budget was calculated as follows:

P = 22,708, S = 100,000

Therefore, T = 22,708/100,000 = 0.22708 seconds.

The total time was calculated for each block by adding all the individual time-required values. The total time required was then rounded off to the nearest minute. An example of some of the entries from the final version of the spreadsheet is shown in Table 4.2. Please refer to Appendix 1 for the comprehensive table of spreadsheet values.

Table 4.2: Example of block 1 entries from data creation spreadsheet

Serial #	ID	Description	Population	Password	Time reqd. in seconds
1	Bas -1-01	One Digit Search	10	9	0.0001000000
2	Bas -1-02	One letter,language specific search	52	O	0.0005200000
3	Bas -1-03	Two digit Search	100	94	0.0010000000
4	Adv -1-01	All one-character, language specific search	256	?	0.0025600000
5	Bas -1-05	Three Digit Search	1,000	173	0.0100000000
6	Bas -1-04	Two letter,language specific search	2,704	gS	0.0270400000
7	Bas -1-07	Four digit search	10,000	3482	0.1000000000
8	Bas -2-17	Dictionary primary search ([EN-1] Common-en-c.adf)	22,708	privs	0.2270800000
9	Bas -2-18	Dictionary primary reverse search ([EN-1] Common-en-c.adf)	22,708	trauts	0.2270800000

For the four blocks consisting of a total of 200 passwords, the time budget calculated is shown in Table 4.3.

Table 4.3: Time budget for each block

Block Number	Budgeted Time
1	10 minutes
2	2 hours 56 minutes
3	22 hours 42 minutes
4	24.8 days
TOTAL AVAILABLE TIME BUDGET:	**7 Days**

As shown in Table 4.3, the budget for block numbers 1, 2, 3, and 4 calculated by the budgeting model formula are 10 minutes, 2 hours 56 minutes, 22 hours 42 minutes, and 24.8 days respectively. The total available time budget as defined by the data requirements was seven days. Thus it was assumed that block numbers 1, 2 and 3 would complete within the overall budget and block number 4 would not be completed in the available time budget.

```
sudo vi /etc/pam.d/common-password


#
# /etc/pam.d/common-password - password-related modules common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define the services to be
# used to change user passwords.  The default is pam_unix.

# Explanation of pam_unix options:
#
# The "sha512" option enables salted SHA512 passwords.  Without this option,
# the default is Unix crypt.  Prior releases used the option "md5".
#
# The "obscure" option replaces the old `OBSCURE_CHECKS_ENAB' option in
# login.defs.
#
# See the pam_unix manpage for other options.

# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules.  See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)
password	[success=1 default=ignore]	pam_unix.so obscure min=1 max=1 md5
# here's the fallback if no module succeeds
password	requisite			pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
password	required			pam_permit.so
# and here are more per-package modules (the "Additional" block)
password	optional	pam_gnome_keyring.so
# end of pam-auth-update config
~
~
~
~
:wq
```

Figure 4.1: Commands typed in Linux shell to change encryption format to MD5

After all the passwords were created, the next step was creating the accounts in the computer running Ubuntu Linux. The computer running Ubuntu was accessed, and it was configured to store passwords in the shadow file using MD5 encryption. The settings for Ubuntu were changed by typing commands in the Linux shell, as shown in Figure 4.1. The command ‘sudo vi /etc/pam.d/common-password’ was first typed in the shell prompt. The file ‘common-password’ is used for storing the operating system settings for password-related services and is restricted to be accessed and modified only by the root account. Since Ubuntu does not allow the operating system to be used by the root user, the ‘sudo’ part of the aforementioned command was used to enable the ‘common-password’ file to be viewed and modified by the root user. The ‘vi’ part of the command indicated the use of the ‘vi’ shell-based text editor for editing the file.

The contents of the file are displayed in the lines following the command. In order to change the settings, the line ‘password [success=1 default=ignore] pam_unix.so obscure min=1 max=1 sha512’ was replaced with ‘password [success=1 default=ignore] pam_unix.so obscure min=1 max=1 md5’. After the changes were made, they were saved by going into the text editor’s prompt by pressing the escape and colon keys followed by ‘wq’ (as shown in the last line of figure 4.1), which is the command for saving the file and exiting the editor.

After the settings of Ubuntu were changed to store passwords in MD5 format in the /etc/shadow file, the various user accounts were created. The user accounts were created with the use of usernames such as main1, which has the password with unique serial number 1 (from the spreadsheet discussed in Section 4.2.2); main2 for the password with unique serial number 2; and thus mainX for the password with unique serial number X. The user accounts were created by accessing the Linux shell and typing the commands displayed in Figure 4.2.

```
vishal@vishal-desktop:~$ sudo adduser main2
Adding user `main2' ...
Adding new group `main2' (1003) ...
Adding new user `main2' (1003) with group `main2' ...
Creating home directory `/home/main2' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for main2
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
vishal@vishal-desktop:~$
```

Figure 4.2: Commands typed in Linux shell to create user accounts

After the user accounts were created, the shadow file was copied to a flash drive using the Linux shell. The copy procedure that was performed in the Linux shell is shown in Figure 4.3.

```
vishal@vishal-desktop:~$ sudo cp /etc/shadow /media/24FA-4543/
```

Figure 4.3: Command typed in Linux to copy shadow password file to the flash drive

The shadow file consisted of all the user accounts and hashes. The hashes are in the Modular Crypt Format (MCF) as described in section 2.1.2. An example of some of the contents of the shadow file is shown in Figure 4.4.

```
main1:$1$/uJtq9Oe$jqlVBDzjYzd4HECJ9vfQx1:14803:0:99999:7:::
main2:$1$Uo.6TWfz$z0wqUFbEJ/FNeOFiUsSBf/:14803:0:99999:7:::
main3:$1$vmO3xBFe$xIF/aeofcoX5obHB0jSse0:14803:0:99999:7:::
main4:$1$OCpN0bmi$xDt/wh0rnZ4BbrihGeA9w.:14803:0:99999:7:::
main5:$1$9X4I4vEk$kskQSltn6pCaPuCrHg8Si1:14803:0:99999:7:::
main6:$1$Nd3K0/wz$Sg.rYj3DrgxL.b3f85r0W0:14803:0:99999:7:::
main7:$1$UEoE4yZk$9WbZGIpCKMO8IqPM.Hom..:14803:0:99999:7:::
main8:$1$9Ixzs6kw$hEJth50OBKJY6iAmqtCdB/:14803:0:99999:7:::
main9:$1$E/L.kqtj$Xh5zQqBzDRjjFiOkIykOX.:14803:0:99999:7:::
main10:$1$juUMc2Gx$A3Slct7Mbp648QR9Ad9/t0:14803:0:99999:7:::
```

Figure 4.4: Example of contents from the shadow file

The various user accounts and hashes were then copied. In order to allow for individual jobs to be created for each user in Distributed Network Attack, the details of each user were stored in one file. Hence, multiple files were created with individual user details. Thus, main1.txt contained ‘main1:1/uJtq9Oe$jqlVBDzjYzd4HECJ9vfQx1:14803:0:99999:7:::’, main2.txt contained its own details and so forth. The next step performed was the creation of individual profiles for each job. The profiles were created in Distributed Network Attack and the profile settings were set to search the population relevant to the password cracking job. The profiles were named ‘1-MAIN’ for the account named main1, ‘2-MAIN’ for the account named main2, and thus X-MAIN for the account named mainX, where X is the unique serial number (as enumerated in the spreadsheet in section 4.2.2). The profiles were created to prepare for the data collection phase mentioned in the following section.

4.2.3 Data Collection and Report Generation

As just discussed, the passwords were created and arranged in ascending order of population size. The passwords were then categorised into groups or blocks of 50, since the AccessData Manual suggests not adding more than 50 files for password cracking at one time. The accounts were then created and shadow file exported. The individual password files containing the user information were also prepared. The profiles for each of the accounts were also created in Distributed Network Attack.

After these preparations, the experiment was started at 8.45 a.m. on 13 July 2010. The first block was added to Distributed Network Attack by adding the individual text files described in Section 4.2.2: main1.txt, main2.txt, etc., up to main50.txt. The files were added one at a time whilst simultaneously assigning them to the respective DNA profiles. Thus, every password had a different start time. The total time required just for adding the files was approximately ten minutes. After the last file was added, the passwords were left for cracking for the budgeted time. The experiment was monitored regularly and a monitoring/action-performed log was maintained. The intervals for monitoring the experiment were either 12 hours or the budgeted time of the block, whichever was earlier. The next block was added after the running block of passwords had finished cracking. Thus, blocks 1, 2, 3, and 4 were added and tested in a similar manner. The contents of the monitoring/action-performed log of the experiment are shown in Table 4.4.

Table 4.4: Monitoring/action-performed log of the experiment

Serial #	Machine Number	Date and Time Monitored	Status / Action Performed	Next Scheduled Monitoring Time
1	10	13/7/2010 - 8:45 AM	Started Adding files - Block 1	
2	10	13/7/2010 - 8:53 AM	Stopped Adding files - Block 1	
3	1,2,3,4,5,6, 9,10	13/7/2010 - 8:55 AM	Machines working and active.	
4	10	13/7/2010 - 8:57 AM	Ended Block 1	
5	10	13/7/2010 - 8:58 AM	Started Adding files - Block 2	
6	10	13/7/2010 - 9:10 AM	Stopped Adding files - Block 2	
7	1,2,3,4,5,6, 9,10	13/7/2010 - 9:13 AM	Machines working and active.	12:06 PM (after 2 hours 56 minutes)
8	10	13/7/2010 - 11:30 AM	Ended Block 2	
9	10	13/7/2010 - 11:31 AM	Started Adding files - Block 3	
10	10	13/7/2010 - 11:44 AM	Stopped Adding files - Block 3	
11	1,2,3,4,5,6, 9,10	13/7/2010 - 11:48 AM	Machines working and active.	13/7/2010 - 8:45 PM
12	1,2,3,4,5,6, 9,10	13/7/2010 - 8:45 PM	Machines working and active.	14/7/2010 - 8:45 AM
13	10	14/7/2010 - 8:45 AM	Ended Block 3	
14	10	14/7/2010 - 8:45 AM to 8:58 AM	Backed up results. Deleted jobs. Restarted Supervisor, due to inability to add more jobs.	
15	10	14/7/2010 - 8:58 AM	Started Adding files - Block 4	
16		14/7/2010 - 9:08 AM	Stopped Adding files - Block 4	
17	1,2,3,4,5,6, 9,10	14/7/2010 - 9:09 AM	Machines Active and working	14/7/2010 - 8:45 PM
18	1,2,3,4,5,6, 9,10	14/7/2010 - 8:45 PM	Machines Active and working	15/7/2010 - 8:45 AM
19	1,2,3,4,5,6, 9,10	15/7/2010 - 8:45 AM	Machines Active and working	15/7/2010 - 8:45 PM
20	1,2,3,4,5,6, 9,10	15/7/2010 - 8:45 PM	Machines Active and working	16/7/2010 - 8:45 AM
21	1,2,3,4,5,6, 9,10	16/7/2010 - 8:45 AM	Machines Active and working	16/7/2010 - 8:45 PM
22	1,2,3,4,5,6, 9,10	16/7/2010 - 8:45 PM	Machines Active and working	17/7/2010 - 8:45 AM
23	1,2,3,4,5,6, 9,10	17/7/2010 - 8:45 AM	Machines Active and working	17/7/2010 - 8:45 PM
24	1,2,3,4,5,6, 9,10	17/7/2010 - 8:45 PM	Machines Active and working	18/7/2010 - 8:45 AM
25	1,2,3,4,5,6, 9,10	18/7/2010 - 8:45 AM	Machines Active and working	18/7/2010 - 8:45 PM
26	1,2,3,4,5,6, 9,10	18/7/2010 - 8:45 PM	Machines Active and working	19/7/2010 - 8:45 AM
27	1,2,3,4,5,6, 9,10	19/7/2010 - 8:45 AM	Machines Active and working	19/7/2010 - 8:45 PM
28	1,2,3,4,5,6, 9,10	19/7/2010 - 8:45 PM	Machines Active and working	20/7/2010 - 8:45 AM

29	1,2,3,4,5,6, 9,10	20/7/2010 - 8:45 AM	Machines Active and working	
30	1,2,3,4,5,6, 9,10	20/7/2010 - 8:45 AM	EXPERIMENT STOPPED.	

The experiment was stopped after seven days, which was at 8.45 a.m. on 20 July 2010. The password cracking reports were then generated using Distributed Network Attack's report-generation feature. As can be seen in the logs, the reports for the first three blocks were generated and saved on 14 July 2010. The report for the fourth block was generated after the experiment was over. A sample of the password cracking report generated by Distributed Network Attack is shown in Figure 4.5. For the entire report, see Appendix 3.

DNA/PRTK Report

C:\Main Test\MainTest-passwdFiles\Block1\main9.txt
Job Status: Finished on 7/13/10 8:46:41
Commonly Registered Type: crypt user: main9
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:46:37
File Modified: 7/13/10 6:49:44
SHA 1: aa5fc73e37b0c1ebf4dc2b7654532a6c32e46fba
MD5: ae9ffdaef80dd967b3aef2749c8e2b49
Result Type:
Result: trauts
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search

C:\Main Test\MainTest-passwdFiles\Block1\main10.txt
Job Status: Finished on 7/13/10 8:46:53
Commonly Registered Type: crypt user: main10
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:46:48
File Modified: 7/13/10 6:49:58
SHA 1: d19b1b30137536ef9be7bc0698140a075e267d44
MD5: 1ed9fa57a9fb389f8cb5c63bf7bef4a8
Result Type:
Result: U%
Description: Unknown
Password Type: Password
Where Found: (ADV-1-02) All two character, language-specific search

Figure 4.5: Sample of DNA's password cracking report

4.3 ANALYSIS OF DATA

The experiment was performed as discussed in Section 4.2.3 and the required data was gathered by generating the cracking report. The cracking report consisted of various entities, as shown in Figure 4.5, including file name and location, job status, commonly registered type, identified type, file size, file version, job started, file modified, SHA 1, MD5, result type, result, description, password type, and where found.

In order to analyse the data, the entities of interest were copied to an Excel spreadsheet. These entities consisted of the username, start time, and finish time. The start time consists of the date and time at which the cracking attempt started on the respective password. The finish time consists of the date and time at which the respective password cracking job was over. Once the values were copied into the spreadsheet, a new column was created to calculate the time taken to crack the password. The time taken for the password to crack was calculated as the difference between the start and the finish times. The 'Time taken to crack' was entered in the 'DD:HH:MM:SS' format, where DD=days, HH=hours, MM=minutes and SS=seconds. Once, the values were entered in the 'DD:HH:MM:SS' format for all four blocks, new columns were created to calculate the total time taken to crack in seconds, in minutes, in hours, and in days as required by the blocks. The analysis performed for each of the individual blocks is described in Sections 4.3.1 to 4.3.4, followed by a section for the time analysis of all the blocks.

4.3.1 Analysis of Block 1

For the analysis of block 1, the time difference between the start time and finish times was calculated in the format of DD:HH:MM:SS. The times taken for the passwords in block 1 were all in the range of a few minutes, since the budget for the first block was about 557.2 seconds or approximately 10 minutes. Thus, the total time taken to crack in seconds was calculated for all accounts in block 1. An example of the analysis performed on the first ten accounts of the block is shown in Table 4.5.

Table 4.5: First 10 accounts' password cracking times for block 1

BLOCK # 1				
USERNAME	START TIME	FINISH TIME	TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)	TIME TAKEN TO CRACK (IN SECONDS)
main1	7/13/10 8:45:09	7/13/10 8:45:11	00:00:00:02	2
main2	7/13/10 8:45:20	7/13/10 8:45:23	00:00:00:03	3
main3	7/13/10 8:45:32	7/13/10 8:45:35	00:00:00:03	3
main4	7/13/10 8:45:43	7/13/10 8:45:47	00:00:00:04	4
main5	7/13/10 8:45:54	7/13/10 8:45:57	00:00:00:03	3
main6	7/13/10 8:46:06	7/13/10 8:46:08	00:00:00:02	2
main7	7/13/10 8:46:16	7/13/10 8:46:20	00:00:00:04	4
main8	7/13/10 8:46:27	7/13/10 8:46:31	00:00:00:04	4
main9	7/13/10 8:46:37	7/13/10 8:46:41	00:00:00:04	4
main10	7/13/10 8:46:48	7/13/10 8:46:53	00:00:00:05	5

As shown in table 4.5, the first ten accounts' password cracking times are all in seconds. The time required to crack each password was first calculated in the DD:HH:MM:SS format, by calculating the difference between the start and finish times. Subsequently, the time taken for password to crack in seconds was then derived. The time required for the remaining 40 accounts' passwords in block 1 are also in the range of a couple of seconds to approximately two to three hundred seconds. For the complete analysis of block 1, please see Appendix 4.

4.3.2 Analysis of Block 2

For the analysis of block 2, the time difference between the start time and finish times was also calculated in the format of DD:HH:MM:SS. The times required for block 2 were mostly in the range of few minutes to less than two hours, since the budget allocated to block 2 was approximately 2 hours and 56 minutes. Therefore, for block 2, the times taken for all accounts were calculated in both seconds and minutes. An example of the analysis performed on the first ten accounts of block 2 is shown in Table 4.6.

Table 4.6: First 10 accounts' password cracking times for block 2

BLOCK#2					
USERNAME	START TIME	FINISH TIME	TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)	TIME TAKEN TO CRACK (IN SECONDS)	TIME TAKEN TO CRACK (IN MINUTES)
main51	7/13/10 8:58:10	7/13/10 8:59:04	00:00:00:54	54	0.9
main52	7/13/10 8:58:22	7/13/10 8:59:23	00:00:01:01	61	1.016666667
main53	7/13/10 8:58:32	7/13/10 9:01:31	00:00:02:59	179	2.983333333
main54	7/13/10 8:58:43	7/13/10 10:26:06	00:01:27:23	5,243	87.38333333
main55	7/13/10 8:58:53	7/13/10 9:06:53	00:00:08:00	480	8
main56	7/13/10 8:59:03	7/13/10 9:12:57	00:00:13:54	834	13.9
main57	7/13/10 8:59:12	7/13/10 9:06:04	00:00:06:52	412	6.866666667
main58	7/13/10 8:59:27	7/13/10 9:50:06	00:00:50:39	3,039	50.65
main59	7/13/10 8:59:37	7/13/10 10:21:44	00:01:22:07	4,927	82.11666667
main60	7/13/10 8:59:47	7/13/10 9:06:20	00:00:06:33	393	6.55

As shown, the times required are in the range of minutes, which have been calculated by converting the time taken to crack in seconds, which was in turn calculated from the time difference between start and finish times calculated in the DD:HH:MM:SS format. The time required for the remaining 40 accounts' passwords from block 2 are also in the range of few minutes to less than two hours. For the complete analysis of block 2, please see Appendix 4.

4.3.3 Analysis of Block 3

Block 3 was analysed by calculating the time taken based on the difference between the start and the finish times. The time differences were calculated in the DD:HH:MM:SS format, in the same manner as the previous blocks. The password cracking time required for accounts in block 3 were mostly in the range of few hours to less than 23 hours. The allocated time for block 3, as mentioned in Section 4.2.2, is approximately 22 hours and 42 minutes. For the analysis, the time taken for each of the accounts to crack was calculated in total of seconds. The total time calculated in seconds was then converted to minutes. After converting the total time required in minutes, the time was then converted to hours. An example of the analysis performed on the first 10 accounts of block 3 is shown in Table 4.7 below.

Table 4.7: First 10 account's password cracking times for block 3

BLOCK#3						
USERNAME	**START TIME**	**FINISH TIME**	**TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)**	**TIME TAKEN TO CRACK (IN SECONDS)**	**TIME TAKEN TO CRACK (IN MINUTES)**	**TIME TAKEN TO CRACK (IN HOURS)**
main101	7/13/10 11:31:48	7/13/10 13:21:33	00:01:49:45	6,585	109.75	1.829166667
main102	7/13/10 11:32:04	7/13/10 16:16:20	00:04:44:16	17,056	284.2666667	4.737777778
main103	7/13/10 11:32:15	7/13/10 16:35:42	00:05:03:27	18,207	303.45	5.0575
main104	7/13/10 11:32:27	7/13/10 18:45:03	00:07:12:36	25,956	432.6	7.21
main105	7/13/10 11:32:42	7/13/10 19:01:19	00:07:28:37	26,917	448.6166667	7.476944444
main106	7/13/10 11:32:53	7/13/10 18:09:00	00:06:36:07	23,767	396.1166667	6.601944444
main107	7/13/10 11:33:05	7/13/10 15:37:26	00:04:04:21	14,661	244.35	4.0725
main108	7/13/10 11:33:16	7/13/10 16:36:11	00:05:02:55	18,175	302.9166667	5.048611111
main109	7/13/10 11:33:30	7/13/10 20:01:49	00:08:28:19	30,499	508.3166667	8.471944444
main110	7/13/10 11:33:41	7/13/10 17:23:28	00:05:49:47	20,987	349.7833333	5.829722222

The first 10 accounts' password cracking times for block 3 lie in the range of 1 hour to less than 12 hours. As above, the time taken to crack was first calculated by calculating the difference between start and finish times, which was then converted to time taken to crack in seconds, then minutes, then hours. The time required for the remaining 40 accounts also lie in the range of 1 to less than 12 hours. For complete analysis of block 3, please see Appendix 4.

4.3.4 Analysis of Block 4

The password cracking times required for block 4 were in the range of 1 to 6 days. The total calculated budget for block 4, according to Section 4.2.2, was approximately 24.8 days. However, the experiment was only run for a total of 7 days due to limited availability of resources. Thus the password cracking procedure for block 4 was unable to be completed, as it was not run for the required budgeted time of 24.8 days. Therefore, for the analysis of block 4, the 24 accounts whose passwords were cracked were considered. The remaining 26 accounts which had not finished processing were discarded from the analysis and results. Block 4 was then analysed by calculating the time differences between the start and the finish times. The time differences were calculated in the

DD:HH:MM:SS format, similar to the ones calculated for the previous blocks. Subsequently, the passwords-cracking times was then calculated in seconds from the time differences in the DD:HH:MM:SS format. The time required in minutes was then derived from the time required in seconds; the time required was then calculated in hours and days in a similar manner. An example of the analysis performed on the first ten accounts of block 4 is shown in Table 4.8 below.

Table 4.8: First 10 accounts' password cracking times for block 4

BLOCK#4							
USERNAME	START TIME	FINISH TIME	TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)	TIME TAKEN TO CRACK (IN SECONDS)	TIME TAKEN TO CRACK (IN MINUTES)	TIME TAKEN TO CRACK (IN HOURS)	TIME TAKEN TO CRACK (IN DAYS)
main151	7/14/10 8:58:56	7/15/10 3:25:21	00:18:26:25	66,385	1106.416667	18.44027778	0.768344907
main152	7/14/10 8:59:14	7/15/10 23:03:12	01:14:03:58	137,038	2283.966667	38.06611111	1.586087963
main153	7/14/10 8:59:25	7/15/10 2:07:12	00:17:07:47	61667	1027.783333	17.12972222	0.713738426
main154	7/14/10 8:59:36	7/17/10 8:02:59	02:23:03:23	255803	4263.383333	71.05638889	2.96068287
main155	7/14/10 8:59:47	7/17/10 15:41:19	03:06:41:32	283292	4721.533333	78.69222222	3.278842593
main156	7/14/10 8:59:59	7/17/10 7:14:26	02:22:14:27	252867	4214.45	70.24083333	2.926701389
main157	7/14/10 9:00:12	7/18/10 7:49:16	03:22:49:04	341344	5689.066667	94.81777778	3.950740741
main158	7/14/10 9:00:23	7/17/10 12:27:54	03:03:27:31	271651	4527.516667	75.45861111	3.144108796
main159	7/14/10 9:00:35	7/19/10 10:04:05	05:01:03:30	435810	7263.5	121.0583333	5.044097222
main160	7/14/10 9:00:45	7/16/10 17:23:46	02:08:23:01	202981	3383.016667	56.38361111	2.34931713

The first ten accounts' password cracking times are in the range of 1 to 6 days. Also, the time difference between the start and finish times, for the accounts shown in table 4.8 are calculated in the DD:HH:MM:SS format. Also shown in table 4.8, the time taken to crack in seconds has been derived from the values of the time taken to crack in DD:HH:MM:SS format. Subsequently, times taken to crack in minutes and hours and days have also been calculated. The remaining 14 accounts also have password cracking times in the range of 1 to 6 days. For a complete analysis of all the 24 accounts, please see Appendix 4.

4.3.5 Time Analysis of All Blocks

After the analyses for all four blocks were completed, the time required for each of the blocks to crack as a whole was considered for analysis. As discussed in

Section 4.2.2, the time budgets were calculated before the main experiment for the blocks as a whole. To allow for comparison, the actual time that each of the blocks required as a whole to complete cracking was calculated.

Each of the individual files in each of the blocks had different start times, since each of the files were added one at a time for cracking (see Section 4.2.3). After the last file was added, the respective block was then left for cracking for the budgeted time. Thus, for analysis, the actual time taken for the entire block to be cracked was calculated as the difference between the start time of the last added file in the block and the finish time of the last password to have cracked in the block. Therefore, the actual time was calculated as the difference between the value of the account with the latest start time in the block (which is the last added file in the block), and the value of the account with the latest finish time in the block (which is the last password to have cracked in the block). To allow for comparison of the results, the time metric of seconds was chosen to compare the actual time and budgeted time of the respective blocks. As explained in Section 4.3.4, block 4 could not complete the cracking procedure. Thus, block 4 could not be considered for time analysis of the whole block, since prior to cracking the time budget was calculated for the whole block. Thus, it would not be possible to compare the actual time taken to crack all of block 4 with its budgeted time. The actual times calculated for the blocks 1 to 3, along with their respective budgeted times, are shown in Table 4.9.

Table 4.9: The actual time and budgeted time for each block

Block Number	Actual Time (in Seconds)	Budgeted Time (in Seconds)
1	210	557.2
2	6749	10556.91
3	42710	81681

As shown in Table 4.9, the actual time taken for block 1 was 210 seconds, compared to its budgeted time of 557.2 seconds. The actual time taken for block 2 to be cracked was 6,749 seconds, compared to its budgeted time of 10,556.91 seconds. Finally, the actual time taken for block 3 was 42,710 seconds, compared to its budgeted time of 81,681 seconds.

4.4 PRESENTATION OF DATA FINDING

After the analysis was performed, bar graphs were created for the purpose of graphical representation. The bar graphs were created for the collected data of blocks 1, 2, 3, and 4 from the analysis of individual blocks done in Sections 4.3.1 to 4.3.4. Also, for visual representation and comparison, bar graphs were created from the time analysis of all blocks performed in Section 4.3.5.

The results for time analysis of the individual accounts in all of the blocks are given in Section 4.4.1. The results for the time analysis of the whole blocks are presented in Section 4.4.2.

4.4.1 Results for time analysis of individual accounts in all blocks

After the analyses of the individual blocks (explained in Section 4.3.1-4.3.4), the results were represented in bar graphs. The results for the time analysis of each of the accounts from each of the blocks are displayed and explained below.

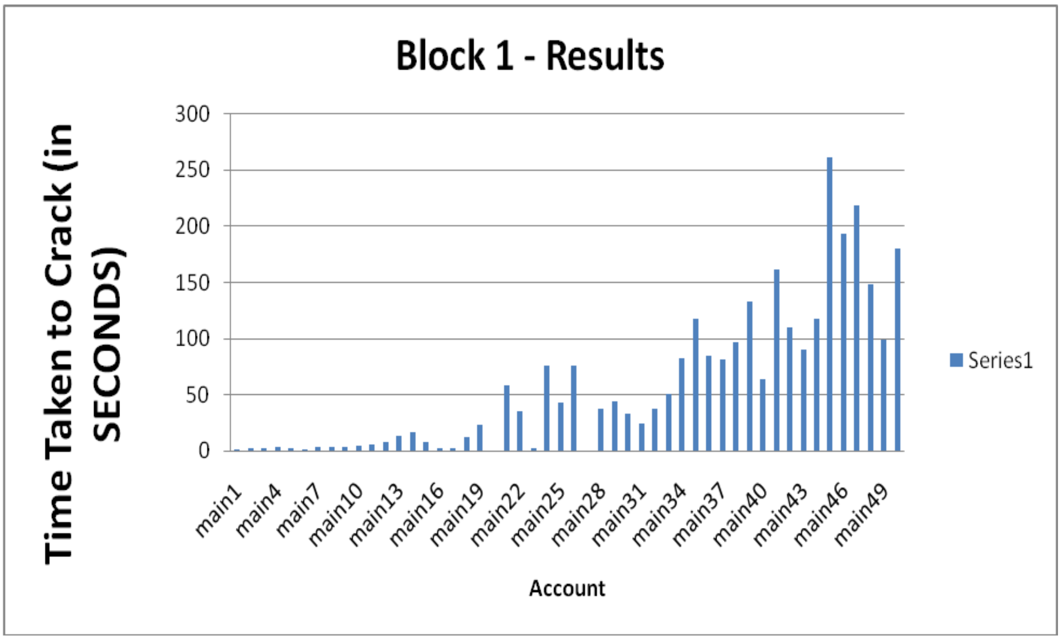

Figure 4.6: Time taken to crack passwords in block 1

Figure 4.6 shows the time taken for all of the accounts in block 1 to be cracked. The X-axis is the name of the account and the Y-axis signifies the time taken for the account's password to be cracked. The graph shown in Figure 4.6 is a visual representation of the analysis table of block 1 (as explained in Section 4.3.1). As shown, the times taken for all of accounts in block 1 to be cracked are in the range of 0 to 300 seconds.

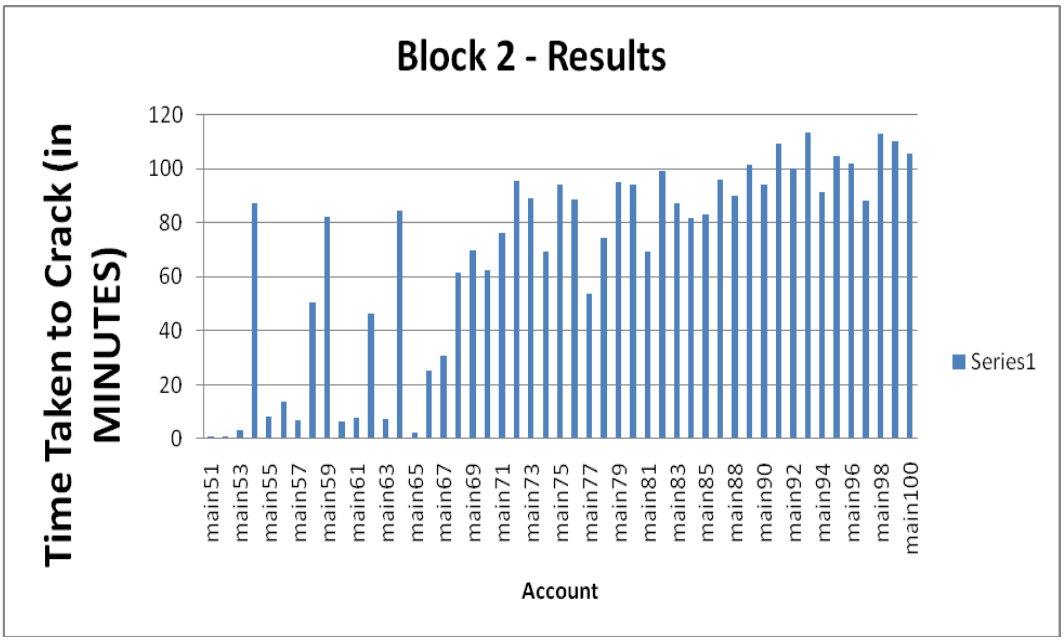

Figure 4.7: Time taken to crack passwords in block 2

Figure 4.7 shows the time taken for all the accounts in block 2 to be cracked. The X-axis is the name of the account and the Y-axis is the time required for the account to crack. The graph shown in Figure 4.7 is a visual representation of the analysis table of block 2 (as explained in subsection 4.3.2). As shown, the times taken for all of the accounts in block 2 to crack are in the range of 0 to 120 minutes.

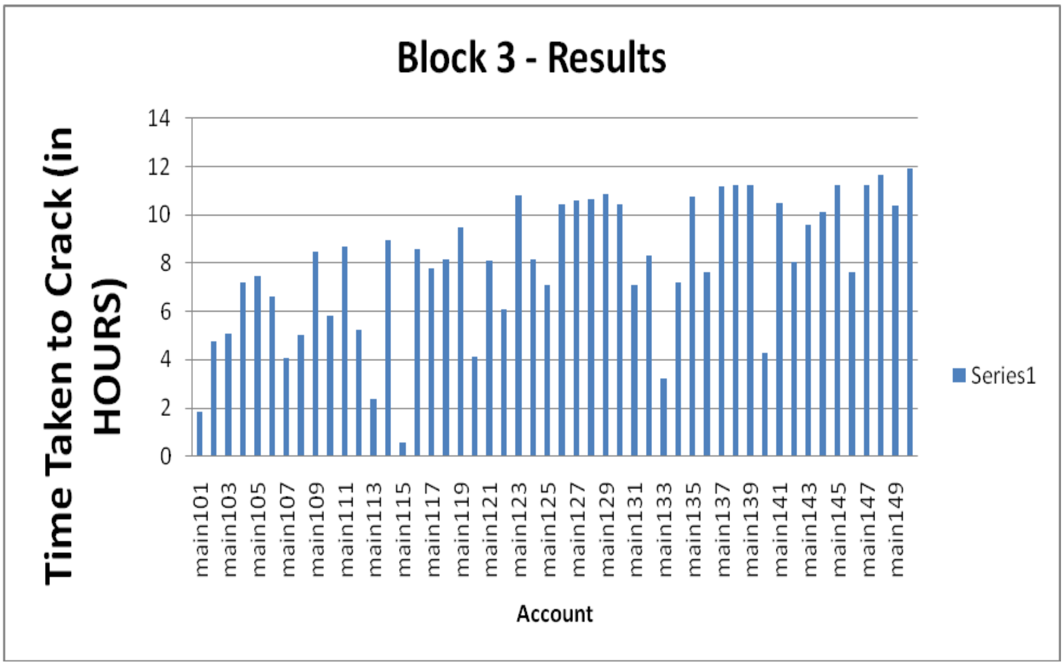

Figure 4.8: Time taken to crack passwords in block 3

Figure 4.8 shows the time taken for all the accounts in block 3 to be cracked. The X-axis consists of the names of the accounts and the Y-axis consists of the time required for the accounts to be cracked. The graph shown in Figure 4.8 is a visual representation of the analysis table of block 3 (as explained in Section 4.3.3). As shown, the times taken for all of the accounts in block 3 to be cracked are in the range of 0 to 14 hours.

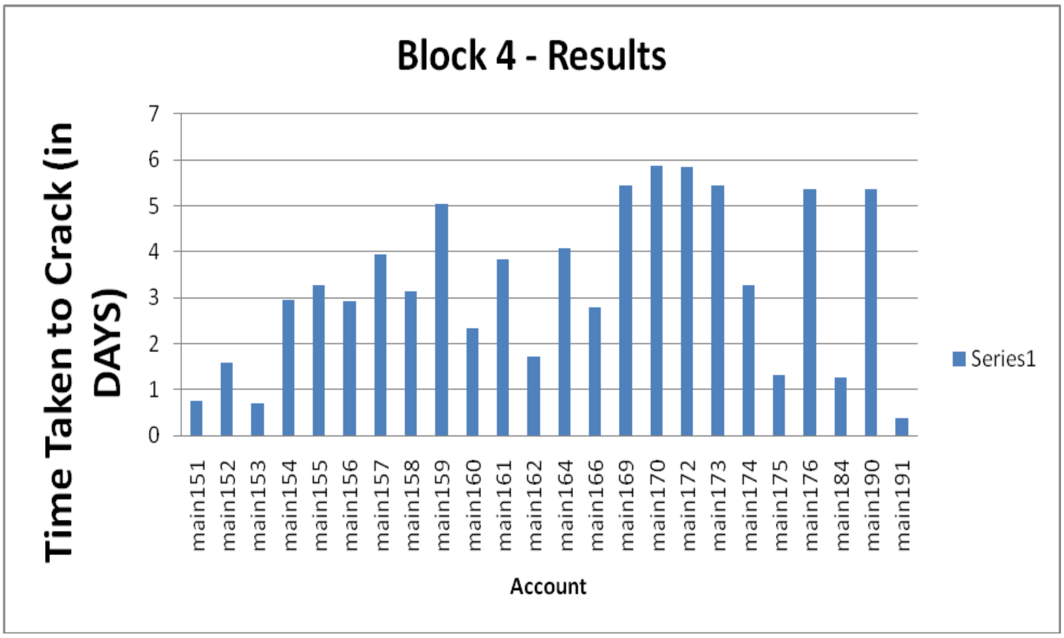

Figure 4.9: Time taken to crack passwords in block 4

Figure 4.9 shows the time taken for all the accounts in block 4 to be cracked. The X-axis consists of the names of the accounts and the Y-axis consists of the time required for the account's respective password to crack. The graph shown in Figure 4.9 is a visual representation of the analysis table of block 4 (as explained in Section 4.3.4). As shown, the times taken for the 24 accounts in block 4 to be cracked are in the range of 0 to 6 days.

4.4.2 Results for Time Analysis of Whole Blocks

After the analyses of the whole blocks (explained in Section 4.3.5) were performed, the results were represented in bar charts. Also, as discussed in Section 4.3.5, it was not possible to perform the time analysis of the whole of block 4. Therefore, the results for block 4 are not considered for graphical representation in this section. The results for the time analyses of the whole blocks numbered 1, 2, and 3 are displayed and explained below.

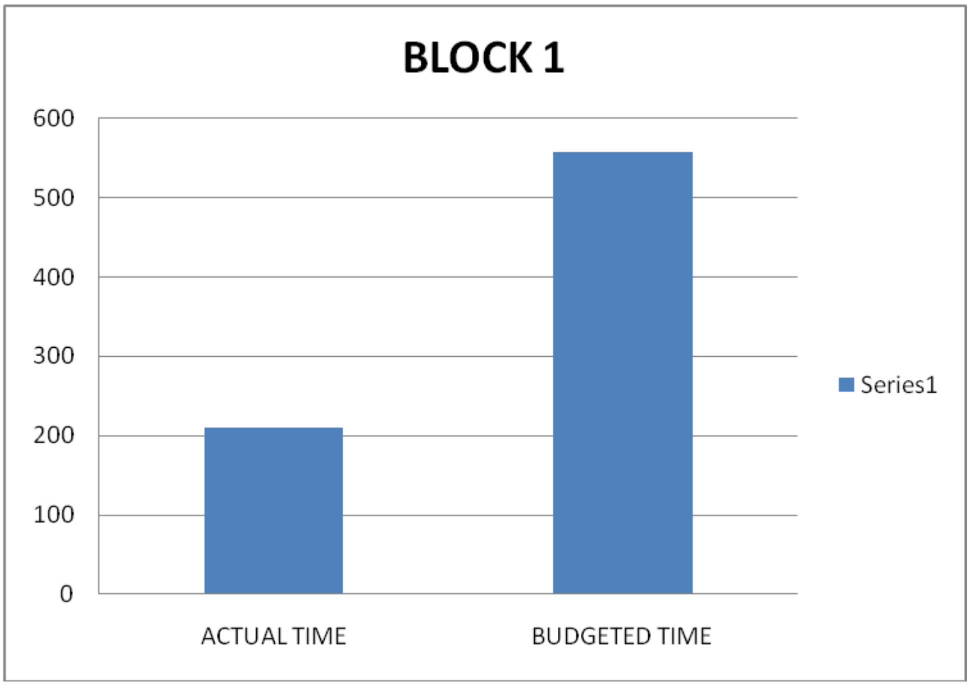

Figure 4.10: Comparison of actual time taken and budgeted time for block 1

Figure 4.10 displays the comparison between the actual time taken and the budgeted time for block 1. The X-axis of the graph displays the labels of actual time and budgeted time, whereas the Y-axis consists of the time required in seconds. The graph in Figure 4.10 is based on the result of analysis from the first row of Table 4.9 in Section 4.3.5. As shown in Figure 4.10, the actual time required for block 1 to be cracked was 210 seconds, whereas the budgeted time for block 1 was 557.2 seconds.

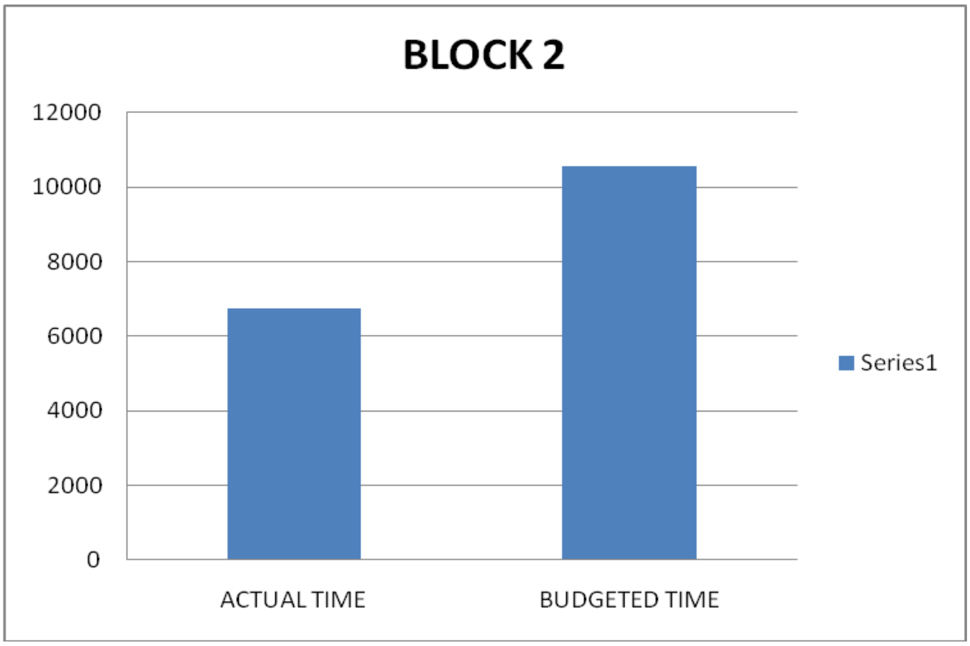

Figure 4.11: Comparison of actual time taken and budgeted time for block 2

Figure 4.11 displays the comparison between the actual time taken and the budgeted time for block 2. The X-axis of the graph displays the labels of actual time and budgeted time, whereas the Y-axis consists of the time required in seconds. The graph in Figure 4.11 is based on the result of analysis from the second row of Table 4.9 in Section 4.3.5. As shown in Figure 4.11, the actual time required for block 2 to be cracked was 6,749 seconds, whereas the budgeted time for block 2 was 10,556.91 seconds.

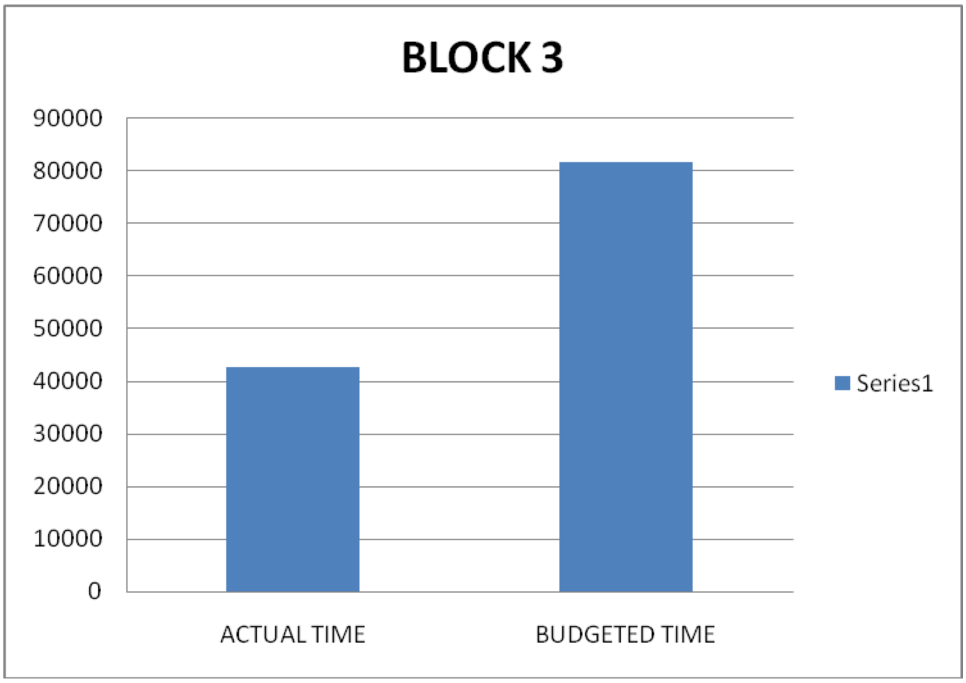

Figure 4.12: Comparison of actual time taken and budgeted time for block 3

The actual time taken and the budgeted time taken for block 3 is compared and displayed in figure 4.12. The X-axis consists of the labels of actual time required and the budgeted time of block 3. The Y-axis consists of the time in seconds. The bar graph in Figure 4.12 is based on the results of analysis from the third row of Table 4.9 in Section 4.3.5. Figure 4.12 displays the actual time required for block 3, that is 42,710 seconds, and the budgeted time of block 3, that is 81,681 seconds.

4.5 CONCLUSION

This chapter reported the deviations from the experimental design and explained all the steps that were taken to perform the research. The steps performed—the four stages of Sampling or Data Creation, Time Budgeting or Data Processing, Testing or Data Collection, and Report Generation and Analysis or Data

Analysis—were explained in detail. The configuration and the experimental setup of the computers used were also detailed in the chapter. For the data creation, the various rules were considered as the population sets. The total of 200 rules were organised in ascending order of population size. A password was then chosen from each of the population sets. For the data processing or time budgeting, the organised password list was divided in groups of 50. The time budgets were calculated for blocks as a whole by the use of the formula presented in Section 3.2.1.4. The account creation procedures and DNA password profile creation procedure were also explained in detail. The experiment was performed and monitored at regular intervals. The DNA report was then generated and considered for analysis.

The analysis of the data was explained in detail in Section 4.3. The analysis included the analysis of individual blocks of passwords as well as time analysis of the blocks as a whole. The data analysed have also been graphically presented in Section 4.4 by means of bar charts. The bar charts were created for the time analysis of the individual accounts in the blocks and also the time analysis of the whole blocks. Thus, the entire research procedure was complete along with the analysis and presentation of data in this chapter. The next chapter discusses and explains the research findings presented in this chapter.

Chapter 5

Research Discussion

5.0 INTRODUCTION

Chapter 4 reported the findings of the laboratory testing; in this chapter, the discrepancy between these findings and what was forecasted in Chapter 2 is reconciled in a discussion. First, the evidence presented in Chapter 4 is used to answer the research question by testing the hypotheses and answering the sub-questions. The findings in Chapter 4 are then discussed with respect to the theory in Chapter 2. The research design as specified in Chapter 3 is also evaluated and discussed with respect to the actual experiment performed as described in Chapter 4.

This research had one main research question and three sub-questions, specified in Section 3.2.3. Based on the experiment performed and the results reported in Chapter 4, the main research question regarding the time implications of cracking passwords using the budgeting model depicted an inconclusive result. Based on the results, it is found that the actual times required to crack the passwords were less than the budgeted times allocated to the blocks of passwords. It is also found that the times required to crack the passwords were very near to half of the budgeted time for each block.

The secondary question regarding the number of passwords cracked in the allocated budget is answered by testing the three hypotheses defined in Section 3.2.3. It is found that the null hypothesis of all of the passwords being capable of being cracked within the time budget allocated by the budgeting model holds true, whereas the other two hypotheses are tested to be false. The remaining secondary questions are also answered based on the results of the experiment. Important matters with regards to recommended budgeting procedures are also discussed in this chapter.

Chapter 5 has been organised as follows. The discussion of research questions along with the answers and the hypotheses tested are found in Section 5.1. The next section, 5.2, discusses the experimental findings and procedures with respect to the theory in Chapter 2 and methodology in Chapter 3. Section 5.3 discusses the budgeting recommendations, based on this research, to be followed by the forensic investigator. The last section, 5.4, contains the concluding remarks.

5.1 DISCUSSION OF RESEARCH QUESTIONS

The main research question is answered in Section 5.1.1. The subsequent sections answer the secondary research questions and give evidence for and against the relevant hypotheses tested in the table sections below.

5.1.1 Answer to the Main Research Question

The main research question, defined in Section 3.2.3, is:

> *Q: What are the time implications of cracking passwords using the budgeting model?*

The research proposed a budgeting model derived from existing best practices. This budgeting model was implemented by means of an experimental design. Based on best practices, the budgeting model consisted of the case, the population set, the password cracking speed, and the total time required to crack the password. Thus, in order to answer the research question, the budgeting model and its implementation must be analysed.

The budgeting model was implemented by a password cracking experiment simulating 200 hypothetical cases. For each of these hypothetical cases, the population set of varying sizes was considered. During the implementation of the budgeting model, certain deviations from the research plan were encountered. (See Section 4.1.) Apart from the deviations encountered in the experiment, there were no other known issues during the implementation of the budgeting model. Thus, the budgeting model was successfully implemented by the simulation of 200 hypothetical cases.

Each of the 200 hypothetical cases had a different population set along with a different password assigned to it. Therefore, based on the simulation results, each of the 200 hypothetical cases was well defined in a unique manner.

Thus, the simulation mirrored the fact that every real-life password cracking case could be unique. For each case, the size of the population set increased in order. Also, all of the passwords (except certain passwords that returned no results) were cracked successfully. Therefore, the population sets chosen for all the passwords (except the passwords that returned no results) were chosen correctly. For the purpose of the password cracking speed, a benchmark value of 12,500 passwords per second per computer was considered, based on a previous study by Bengtsson (2007). Thus, considering the password cracking speed, and the size of the population of each case, the total time required to crack each password was calculated. The total time required to crack each password was calculated by the budgeting model formula:

> Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

The time budgets were allocated to groups or blocks of 50 passwords each (see Section 4.2.2). The budgets were calculated by summing the budgets of the individual accounts (see Section 4.2.2). The individual budgets of all the accounts in each block were then added together to form the total budget for the block. While the individual accounts in blocks 1, 2, 3, and 4 have been analysed (see Sections 4.3.1-4.3.4, 4.4.1), nevertheless, for the main experiment, the budget of the entire block is considered. Therefore, the times required to crack the entire block of passwords will be considered to answer the main research question.

All of the accounts in blocks 1, 2 and 3 were cracked within their allocated time budgets (see Sections 4.3-4.4). The time budget allocated to block 4 was outside the scope of the overall available time budget of seven days. Therefore, the fact that block 4 did not complete can be considered an estimation that was made as per the budgeting model. Also, as per the results mentioned in Sections 4.3 and 4.4, the actual times required were less than the budgeted time. It can be observed from the time analyses of the whole blocks (presented in Sections 4.3.5 and 4.4.2) that the actual time required for block 1 was 210 seconds, compared to the budgeted time of 557.2 seconds. The actual time for block 2 was 6,749 seconds, compared to the budgeted time of 10,556.91 seconds. Lastly, the actual time for block 3 was 42,710 seconds, compared to the budgeted time of 81,681 seconds.

The results for each of the blocks signify that the actual time required to crack the passwords is very near to half of the budgeted time. Since such a trend has been observed, it can be proposed that the correct time budget would have been more accurate if it had been set near to half of the existing time budget. Due to the observed difference between the budgeted time and actual times, it would be fair to say that further research is required to accurately answer the research question. Hence, based on this research study, the time implications based on the budgeting model depict an inconclusive result. As a result, the main research question can be answered in the following manner:

> *A: Based on the research findings, the times allocated to the blocks of passwords depict an inconclusive result. The findings signify that the actual times required to crack the passwords are very near to half of the existing allocated time budgets.*

5.1.2 Sub-questions and Hypotheses Tests

The first secondary research question, as enumerated in Section 3.2.3, is:

> *Q1: How many passwords can be cracked within the allocated time budget?*

To answer this research question, the associated hypotheses H0, H1 and H2 are tested, as shown in Tables 5.1, 5.2 and 5.3. The evidence for the tests is extracted from Chapter 4 results.

Table 5.1: Testing hypothesis H0

<table>
<tr><td colspan="2">Hypothesis H0:
All of the passwords can be cracked within the time budget allocated by the use of the budgeting model.</td></tr>
<tr><td>ARGUMENT FOR:

As shown in the results in Appendix 3 for blocks 1, 2, 3, and 4, the number of successfully cracked passwords are 44, 40, 41, and 20, respectively. Also, based on the results of the study, the number of passwords to have returned no results in blocks 1, 2, 3 and 4 are 6, 10, 9, and 4, respectively. For the context of this study, the password not being returned can be viewed as comparable to the password being successfully cracked in the last attempt utilising the maximum amount of time. Thus, for the context of this study, for blocks 1, 2 and 3, all of the given passwords were cracked within the given budget for the respective blocks.</td><td>ARGUMENT AGAINST:

Based on the results shown in Appendix 3, there were a certain number of passwords in blocks 1, 2, 3, and 4 to have returned no passwords. If in the context of this study, the ‘no password found’ results are not considered as the password being found in the last attempt, then it could be said that only some of the passwords for blocks 1, 2, 3, and 4 were cracked.</td></tr>
<tr><td colspan="2">SUMMARY:
The ‘argument for’ states the reasons for considering the Hypothesis H0 true for blocks 1, 2 and 3 if, in the context of this study, the ‘no password found’ results are considered equivalent to the password being found in the last attempt. The Hypothesis H0 is indeterminate for block 4, since block 4 could not run for the allocated budgeted time. The ‘argument against’ state the reasons for considering H0 to be false for blocks 1, 2, 3, and 4.
For the context of this study, the ‘no password found’ results are being considered equivalent to the password being found in the last attempt. Thus, based on the results of this study, the hypothesis H0 that ‘All of the passwords can be cracked within the time budget allocated by the use of the budgeting model’ does hold true.</td></tr>
</table>

Table 5.2: Testing hypothesis H1

<table>
<tr><td colspan="2">Hypothesis H1:
None of the passwords can be cracked within the time budget allocated by the use of the budgeting model.</td></tr>
<tr><td>ARGUMENT FOR:

Based on the results shown in Appendix 3 and 4, there are no arguments to support the hypothesis that ‘None of the passwords can be cracked within the time budget allocated by the use of the budgeting model’.</td><td>ARGUMENT AGAINST:

Based on the results shown in Appendix 3 and 4, 44 passwords from block 1 were cracked while 6 accounts returned no passwords. For blocks 2, 3, and 4, the numbers of passwords cracked were 40, 41 and 20 respectively while the numbers of accounts returning no passwords were 10, 9, and 4, respectively. Thus, based on the results, every block has nonzero number of passwords that were cracked.</td></tr>
<tr><td colspan="2">SUMMARY:
Based on the results of this study, there are no arguments to support hypothesis H1. A certain number of passwords were cracked in every block. Thus, the hypothesis H1 that ‘None of the passwords can be cracked within the time budget allocated by the use of the budgeting model’ does not hold true.</td></tr>
</table>

Table 5.3: Testing hypothesis H2

Hypothesis H2: *Some but not all of the passwords can be cracked within the time budget allocated by the use of the budgeting model.*	
ARGUMENT FOR:	**ARGUMENT AGAINST:**
Due to the deviations from the research design encountered in the experiment (presented in Section 4.1), some passwords returned no results. For blocks 1, 2, and 3, the number of accounts that returned no passwords and had the job status 'Finished' were 6, 10, and 9 respectively. For block 4, the number of accounts that returned no passwords and had the job status 'Finished' was 4. Also, 26 passwords from block 4 still had not finished being cracked, since it did not run for the entire budgeted time. Thus, since the passwords in blocks 1, 2 and 3 were not actually 'cracked', it could be fair to say that some of the given passwords were cracked for blocks 1, 2 and 3 while others were not. Also, block 4 did not run for the entire budgeted time, and still had accounts that returned no results. Thus, it could also be fair to say that some but not all of the given passwords were cracked for block 4.	As displayed in Appendix 3 and 4, some passwords in each of the blocks returned no results. The job status of the accounts that returned no results was 'Finished', which indicates that the entire population set was searched in order to crack the password. Thus, the accounts that returned no results could be seen as comparable to accounts that returned results in the last attempt, since in both the scenarios the entire population set is scanned and the maximum time is utilised. The reason for the accounts returning no passwords is unknown. It is outside the scope of this study to investigate the cause of the accounts returning no password. Therefore, if the accounts that returned no results are considered comparable to accounts that returned results in the last attempt, for the context of this study, it could be said that the entire blocks 1, 2, and 3 were cracked.
SUMMARY: The 'argument for' state the reasons for considering hypothesis H2 to hold true for blocks 1, 2, 3 and 4. The 'argument against' state the reasons for considering H2 to hold false for blocks 1, 2 and 3 only. If the 'no password found' results in block 4 are considered equivalent to the password being found in last attempt, H2 would be considered indeterminate, since the experiment did not run for the time budgeted for block 4. Thus, based on the context of this study, it can be said that all of the passwords for blocks 1, 2 and 3 were cracked. Therefore, based on the results of this study, it can be concluded that hypothesis H2 is false.	

The hypotheses H0, H1, and H2 have been tested as shown above in Tables 5.1, 5.2 and 5.3 respectively. The arguments supporting each hypothesis and the arguments against each hypothesis are listed, with conclusions drawn from the arguments provided in the summary. Based on the tests of the hypotheses, it has been found that the hypothesis H0 holds true. It has also been found that the hypotheses H1 and H2 do not hold true. Therefore, this research question can be answered in the following manner:

> *A1: For the current experiment, all of the 50 passwords in blocks 1, 2, and 3 were cracked successfully within the allocated budget. Therefore, based on the results of the experiment, it has been demonstrated that all of the given passwords can be cracked within the time budget allocated by the use of the budgeting model.*

The next secondary research question, as defined in Section 3.2.3, is:

> *Q2: How much time is required to crack all of the passwords in each of the given blocks?*

As noted in Section 4.3.5, the time required to crack all of the passwords in block 1 was 210 seconds. The time required to crack all of the passwords in block 2 was 6,749 seconds. The time required to crack all of the passwords in block 3 was 42,710 seconds. Block 4 could not be considered in calculating the total time required to crack all of the passwords, since it was not run for the budgeted time (see Section 4.3.5). As a result, Block 4 could not complete cracking and not all of the passwords in block 4 were cracked. Therefore, this research question can be answered in the following manner:

> *A2: Based on the results of this study, the amount of time required to crack all of the passwords in blocks 1, 2, and 3 are 210, 6,749 and 42,710 seconds, respectively. The amount of time required to crack all of the passwords in block 4 is unknown.*

The next secondary research question, as mentioned in Section 3.2.3, is:

Q3: What are the guidelines for best-practice advice for digital forensic investigators in the field of cracking passwords using AccessData Distributed Network Attack (DNA)?

The existing password recovery strategies using Distributed Network Attack have been detailed in Section 2.1.4. The results of the experiment are given in Appendices 3 and 4. Also, as highlighted above, the hypothesis H0 that 'All of the passwords can be cracked within the time budget allocated by the use of the budgeting model' holds true. The budgeting model has been successfully implemented within the declared constraints and limitations. Thus, it could be fair to say that based on the constraints of, and by the use of, the budgeting model, it could be possible to budget time and allocate resources for password cracking cases. However, as demonstrated in Section 5.1.1, the time implications for the budgeting model are inconclusive. Therefore, a budget allocated using the budgeting model would not be an accurate representation of the actual amount of time required to crack a password. Hence, the budgeting model guidelines could be suitable to be followed as best practice when used in conjunction with the existing password recovery strategies and also in conjunction with its limitations. For an in-depth explanation, refer to Sections 5.2 and 5.3. Thus, this research question can be answered in the following manner:

A3: Based on the results of the study, all of the passwords were cracked within the given budget. Thus, the budgeting model was successfully implemented and operationalised within its given constraints and limitations. The hypothesis that 'All of the passwords can be cracked within the time budget allocated by the use of the budgeting model' has also been demonstrated to be true. However, the budget allocated using the budgeting model would not be an accurate representation of the actual amount of time that may be required to crack a password. Therefore, while the budgeting model guidelines are suitable to be followed as best-practice advice for password cracking using Distributed Network Attack, the constraints and limitations of the budgeting model must also be considered and it should be used in *conjunction with existing password-recovery strategies and best practices.*

5.2 DISCUSSION OF FINDINGS

The research questions have been answered and the relevant hypotheses tested in the previous section. Further discussion of the findings with respect to the theory as defined in Chapters 2 and 3 is presented in this section.

Section 5.2.1 consists of the discussion of time analysis of individual blocks, while section 5.2.2 discusses the time analysis of whole blocks.

5.2.1 Discussion of Time Analysis of Individual Accounts in All Blocks

The time analysis of the individual accounts in all blocks was presented in Section 4.3. For the entire list of results from analysis, see Appendix 4. The findings have also been presented in Section 4.4. From Figures 4.6 through 4.9 in Chapter 4, it can be seen that the times required to crack each of the accounts in each of the blocks were variable. As per the theory presented in Section 2.4.2.3, the password may be cracked in the first attempt or any number of attempts after it. It may also take the maximum number of attempts available to crack. This is why each of the accounts was cracked with a different number of guesses. As a result, the time required for each of the accounts is also variable. Furthermore, Distributed Network Attack has its own job-scheduling algorithm and workload-distribution algorithm to distribute the password cracking workload to the worker computers. Thus, at certain times, some passwords are being processed for cracking whilst other passwords in the block are put on hold. This leads to further variation of timings for the cracking of individual account passwords in each of the blocks.

The passwords were arranged in increasing order of population size (see Sections 3.3.1 and 4.2.2). After sorting the passwords, the time budgets for each of the blocks were calculated as described in Section 4.2.2.

The time budgets for each of the passwords were calculated by the defined budgeting model formula:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

Since the passwords were arranged in increasing order of population size, the time budgets calculated were also in increasing order of size. The budgeted time for block 1 was 10 minutes, block 2 was 2 hours 56 minutes, block 3 was 22 hours 42 minutes, and block 4 was 24.8 days. As mentioned above, based on the results of the experiment (see Appendices 3 and 4), all the passwords were cracked within the times allocated by the budgeting model. Thus, for each block, the times taken for cracking were in increasing order. As seen in Graphs 4.6,-4.9, the times taken for accounts in blocks 1, 2, 3, and 4 are in seconds, minutes, hours, and days respectively. Hence, the findings indicate that using the budgeting model and experimental design made it possible to allocate accounts within blocks and thus allocate resources of time accordingly. Therefore, in real-life forensic investigations, use of the budgeting model (within its limitations), would make it possible to allocate resources of time for password cracking assignments.

5.2.2 Discussion of Time Analysis of Whole Blocks

In the time analysis of whole blocks (Section 4.3.5), the actual time taken to crack the whole block of passwords has been calculated. As shown in Table 4.9, the actual time required for block 1 was 210 seconds compared to its budgeted time of 557.2 seconds. The actual time required for block 2 was 6,749, compared to its budgeted time of 10,556.91 seconds. The actual time required for block 3 was 42,710 seconds, compared to its actual time of 81,681 seconds. A time analysis of block 4 was not conducted since the entire block was not cracked. As per the budgeting model, the budgeted time required for block 4 was approximately 24.8 days. Since the actual experiment was budgeted to run for only a total of 7 days (see Section 3.2.2), it was expected that block 4 would probably not have been cracked in a budget less than 24.8 days. The results in Section 4.3.4 demonstrate that, as expected, block 4 did not complete. If more time were given for the experiment to be completed (preferably the entire budgeted time for block 4), theoretically, the entire block 4 would have been cracked.

The actual time required to crack each of the blocks was approximately half of the budgeted time (see Table 4.9 and Figures 4.10-12). This is a very important result, since it shows that the actual time required was much less than the budgeted time. There could be many reasons for such a result. One reason is that the actual password cracking speed may have been different from the benchmark speed (defined in Section 3.3.2). This fact, however, is not verifiable, since it was not possible to record password cracking speeds for the whole blocks accurately.

Another reason for the actual time being very near to half of the budgeted time could be that the password may be cracked in the first attempt or any number of attempts until the last attempt (as explained in Section 2.4.2.3). If the password were cracked in the first attempt, the amount of time required would be minimal. If the password were cracked in the last attempt, the amount of time required would be maximal. The budgeted time in the budgeting model (explained in Section 3.2.1) budgets the time required to crack a password based on the maximum time that a password might require to be cracked. Also, theoretically, it could be said that the password would on average be cracked in half of the number of required attempts. Due to this, the results could have displayed the trend of the actual password cracking time being very near to half of the budgeted time. Thus, on an average, the passwords may require half of the budgeted time to crack.

The results also show an improvement on the formula that could be tested in future research. Future research may include researching the use of the budgeting model formula of:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S)) / 2.

Thus, the results also imply that in real life forensic investigations, on an average, the password may require half of the budgeted time to crack calculated by the budgeting model.

5.2.3 Research Design Evaluation

The research experiment consisted of a simulation of many hypothetical password cracking cases. This subsection evaluates the research design and performance by comparing the specifications in Chapter 3 with the actual experimental processes as shown in Chapter 4. The evaluation of the research design will give insights into the processes carried out, thus helping improve processes and adding to the existing best-practice knowledge for forensic investigators. The research consisted of four phases, namely Data Creation, Data Processing, Data Collection and Data

Analysis (see Section 3.2.3). All four phases carried out in the experiment are evaluated below.

The Data Creation phase involved creating a sample of passwords from the relevant population sets (see Sections 3.3.1 and 4.4.2). Therefore, it is not essential to evaluate the specifications that were defined for the data creation phase. Also, as per the experimental process carried out, all of the passwords were successfully cracked. Certain accounts returned ‘no passwords’ for unknown reasons. An investigation of this issue was not carried out since it was outside the scope of this study; however, there could be several possible explanations for such an outcome. One explanation could be sampling error, possibly due to insufficient documentation. Another reason could be a software issue with Distributed Network Attack. If the reason no passwords were found is considered to be sampling error, it demonstrates the importance of correctly mapping the correct password to the correct population set. Therefore, in real-life forensic investigation, it is essential that the forensic investigator is able to map the correct relevant information, based on the case, in order to increase the chances of successfully recovering the password.

In the Data Processing phase, a total of 200 passwords were sorted and grouped in blocks of 50. Also, the time budgets for each of the blocks were calculated (see Sections 3.3.2 and 4.2.2). As per the results and the hypotheses tested above, all of the given passwords were cracked successfully in the given time budgets. Therefore, the password sorting and grouping was a successful process used to categorise password cracking jobs. Also, the time budgeting procedure as per the budgeting model was successful in its implementation.

In the Data Collection phase, the passwords were added to Distributed Network Attack whilst assigning them to their respective profiles. The passwords were then kept for cracking for the budgeted time whilst being monitored regularly (see Section 3.3.3 for specifications, 4.2.3 for data collection procedure, and 4.1.3 for the deviations from research plan). The data collection process was carried out as per the specifications with certain deviations as well. As discussed in the monitoring and action-performed logs in Section 4.2.3, there were certain computer performance issues. Due to these issues, the results had to be backed up and the supervisor computer restarted after the cracking process for Block 3 was completed. The computer performance issues were not investigated, since they were outside the scope of this research. However, a possible explanation for such computer performance issues could be based on the number of passwords in each block. As discussed in Section 3.3.1, 50 passwords were chosen in each block, since that was the maximum number recommended by AccessData Corp (2010) in order to maintain computer system and software performance. Thus, it is a possibility that there could have been computer performance issues because the maximum recommended number of 50 passwords was chosen for each block. Therefore, if the computer performance issues were due to the number of passwords in each block, based on the study, it would be recommended to lower the number of passwords in each block. Hence, for example, there could be 30 passwords in each block instead of 50.

Another possible reason for performance issues may be the completed jobs being left in the queue. Therefore, after each block is completed, it is recommended to save and back up the results and clear the job queue before adding new jobs in Distributed Network Attack. Also, as discussed in Section 4.2.3, since it was not possible to monitor the experiment at all times, the experiment was monitored at regular intervals of 12 hours or the budgeted time, whichever was the earliest. Also, since each and every file in every block had to be added manually (see Section 4.2.3), if any block finished cracking earlier than the budgeted time, the computers were left idle until the next scheduled monitoring time. Thus, due to technical limitations, time was not efficiently utilised. Such a problem could be overcome if the experiment were monitored at all times. However, in real-life scenarios, such an action may not always be practically possible. Also, such an action would increase labour costs. Another alternative could be software solutions capable of automating the monitoring process. If such solutions were used in real-life scenarios, for every block, time could be utilised in a much more efficient manner. Also, monitoring the cracking procedure regularly also proved to be of importance since it was possible to trace issues and take corrective action. Therefore, for real-life forensic or password-recovery cases, it would be good practice to monitor the password cracking procedure regularly.

The Data Analysis procedure (discussed in Section 4.3) was helpful to observe the time trends. The results of the analysis provide useful insights for the forensic investigator. However, the procedure followed is not relevant or required information for forensic investigators to follow in real-life cases. Therefore, it is not necessary to evaluate the data analysis procedure. For an in-depth discussion of results, see Sections 5.2.1 and 5.2.2.

5.3 DISCUSSION OF RECOMMENDATIONS

The discussion of findings and the research design evaluation were carried out in the previous section. This section discusses recommendations that may be useful for digital forensic investigators to follow in real-life scenarios.

Based on the results of the study, the budgeting recommendations for forensic investigators are described in section 5.3.1.

5.3.1 Budgeting Recommendations for Forensic Investigators

Based on all of the work done in this research, the budgeting recommendations for forensic investigators are discussed in this section. For digital forensic investigations, evidentiary data may be present in encrypted form in various elements of the system, such as files or elements of spreadsheets and databases. In these cases, it is necessary to crack the password to gain access to the required evidence. The budgeting recommendations and procedures based on the results of the study can be organised in the steps shown in Figure 5.1.

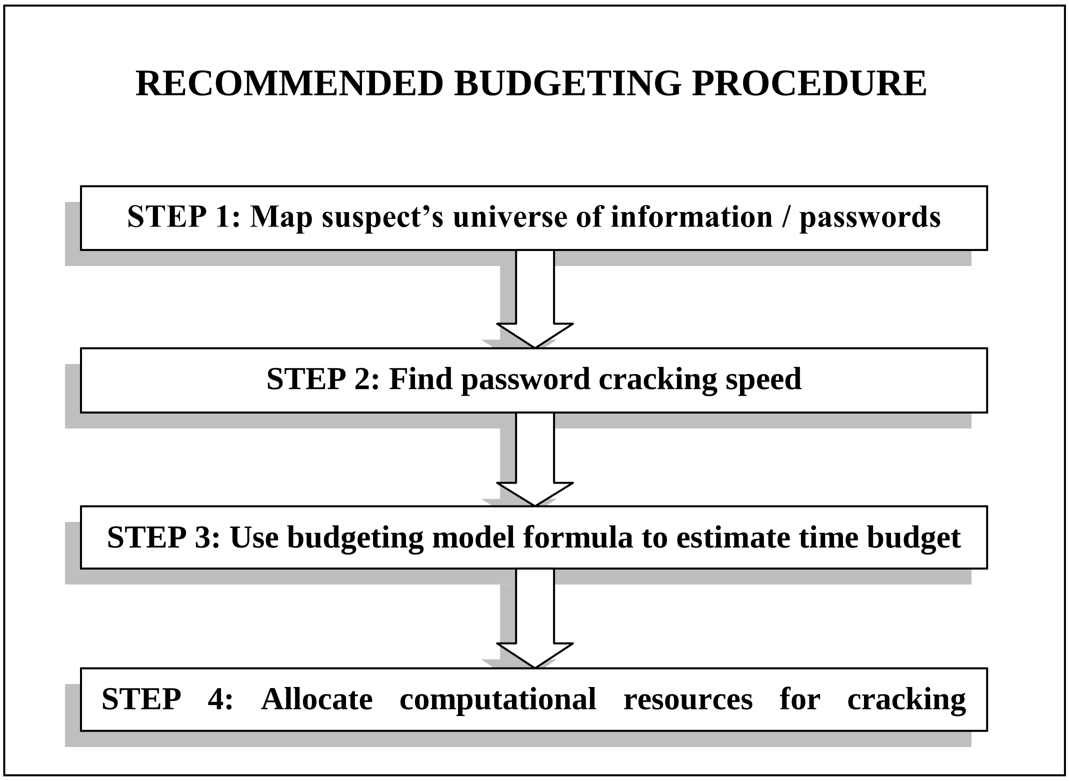

Figure 5.1: Recommended budgeting procedure

Each of the steps displayed in Figure 5.1 is explained in detail in the subsections below. Section 5.3.1.1 explains the recommendations for the first step, which is to map the suspect's universe of information/passwords. Section 5.3.1.2 explains the next step, which is to find the password cracking speed. The next step of using the budgeting model formula is described in Section 5.3.1.3. The last step of allocating computational resources for cracking has been explained in Section 5.3.1.4.

5.3.1.1 First Step – Map Suspect's Universe of Information/Passwords

The first step is to follow the password recovery strategies listed in Section 2.1.4 and 'map' the suspect's universe of information/passwords. Thus, it is important to discover all the essential information with regards to the suspect. Information like the languages spoken by the suspect, the languages and codepage or keyboard settings supported by the suspect's computer, and owner's biographical information are important clues that may be used to improve chances of discovering the suspect's password (AccessData Corp, 2006). Other clues to look out for include handwritten notes and other documentation. Such clues can be converted to digital form using character recognition and can be exported to wordlists.

The suspect's drives can also be imaged and then converted into wordlists. The wordlists created could then be used in the password cracking program (such as Password Recovery Toolkit or Distributed Network Attack) in order to recover the passwords. Criminal profiling techniques may also be used to discover the suspect's key characteristics, thus helping the forensic investigator to identify the nature of the suspect and the nature of the passwords the suspect may be capable of having. In this manner, the forensic investigator could 'map' the suspect's universe of information/passwords. The suspect's universe of passwords should then be converted into wordlists or dictionaries, which can then be considered the 'population set' to be used to attempt to crack the suspect's password using a password cracking program such as Distributed Network Attack or Password Recovery Toolkit.

5.3.1.2 Second Step – Find Password Cracking Speed

Once the suspect's universe of information/passwords has been identified, the next step recommended for digital forensic investigators is to determine the password cracking speed available. Password cracking speed is the average number of passwords attempted per second in order to crack the password (see Section 3.2.1.3). There are various factors that affect the password cracking speed, such as the computer's speed, amount of available memory, number of machines, and encryption strength.

There are many ways to determine password cracking speed. One method would be to run practical tests of passwords encrypted in the same algorithm using the same machines. The average speed could then be noted down and used as a benchmark password cracking speed. The number of computers used may be one or many. If more than one computer is to be used for the case, and if all of the computers have the same configuration, the benchmark tests may be carried out on one computer. Thus, the benchmark speed for one computer can be used, and later on whilst allocating resources (in step four, Section 5.3.1.4), the speed may be multiplied by the number of computers. If all of the computers to be used do not have the same configuration, then it is possible to carry out the benchmark tests on all of the computers the investigator wishes to use. In this manner, the investigator may gain the password cracking speed for the combined use of the entire computing cluster. Thus, if the speed for the entire cluster is calculated, and it has been decided that the entire cluster would be used for the password cracking case, the investigator need not multiply the password cracking speed by the number of computers (as shown in Section 5.3.1.3).

The digital forensic investigator must note and utilise the password cracking speed accordingly as per requirements. For example, all the computers may have the same configuration and the password to be cracked may be encrypted in the MD5 format. The forensic investigator may create three sample passwords in that format. The investigator may then attempt to crack the three passwords on one computer, noting down the password cracking speed for each. If for example, if the speeds noted down are 12,000, 12,500, and 13,000 passwords per second in the three cracking attempts, then the average password cracking speed would be:

(12000 + 12500 + 13000) / 3 = 12500 passwords / second

The forensic investigator may then use 12,500 passwords per second as the benchmark value in the budgeting model formula described in the next section. Whatever the average password cracking speed is determined to be, it can be used in the budgeting model formula to estimate the budget.

5.3.1.3 Third Step – Use Budgeting Model Formula to Estimate Time Budget

After determining the average password cracking speed available, the next step is to use the obtained benchmark value in the budgeting model formula. The formula helps determine the estimated time budget required to crack the password. The budgeting model formula, as described in Section 3.2.1.4, is:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

The investigator must also decide how many computers are required for the password cracking case. The more computers, the faster the speed of password recovery. The computers to be used could be connected in a network and a clustered password cracking program such as Distributed Network Attack could be used. Once the number of computers in the cluster has been decided, the investigator may then multiply the password cracking speed by the number of computers to be used. However, all of the computers must be of the same configuration, since the password cracking speed achieved on each computer of the same configuration would be the same.

For example, if the total size of the population set obtained in step 1 (Section 5.3.1.1) is 12,500,000 words, the benchmark password cracking speed is 12,500 passwords per second for one computer, and the total number of computers to be used is four, then the total time required to crack a password would be calculated as:

Total time required for password to crack (T) = (Total Population Size (P)/ Password Cracking Speed (S))

Therefore, T (in seconds) = 12,500,000 / (12,500*4)

= 12,500,000 / 50,000

Therefore, T = 250 seconds = 4.16 minutes.

Thus, to run the wordlists of 12,500,000 words in the password cracking program, at the average speed of 50,000 passwords per second (12,500 passwords per second for each of the four computers), the maximum time required for the password to be found would be 4.16 minutes.

The practical limitation of having the correct password present in the word list does exist. However, if the word list has been accurately and well chosen, and if the correct password does exist in the chosen population set, the maximum time required would be 4.16 minutes. Therefore, the time budget can be calculated in the aforementioned manner. Thus, once the time budget is estimated, it may be possible to determine the other costs involved for cracking a password.

5.3.1.4 Fourth Step – Allocate Computational Resources for Cracking

After completing the recommended budgeting procedure steps 1 to 3, the last step is to allocate the computational resources for cracking. The passwords should then be kept for cracking on the computers for the allocated time budget specified in Section 5.3.1.3.

There are several possible outcomes of a password cracking case. One outcome could be finding the correct password within the specified time budget. On average, the time required would be half of the specified time budget. Another outcome could be failing to find the password after having attempted all the passwords in the entire population set. In such a case, the investigator may choose to revise the population set or seek alternative means to gain access to the encrypted evidence.

5.4 CONCLUSION

The main research question and all of the secondary research questions have been discussed and answered based on the results provided in Chapter 4. The findings have indicated that the actual times taken to crack the passwords are less than the times allocated to the blocks by the use of the budgeting model. It has also been found that the times required to crack each of the blocks are very near to half of the budgeted time. The hypotheses tests have indicated that all of the passwords are capable of being cracked within the time budgets calculated using the budgeting model. The experimental performance indicated that the budgeting model was successfully implemented and operationalised within its given constraints. Thus, it was demonstrated that the budgeting model can be used as best-practice advice for forensic investigators in the field of password cracking. However, the constraints and limitations must be taken into consideration and it should also be used in conjunction with the existing best practices.

The results of the time analysis of the individual accounts in all blocks, and of the blocks collectively, have been discussed. As a result of the discussion of the time analysis of whole blocks, it has been found that the budgeting model formula can be revised and retested in future research by considering half the calculated time budget. The research design has also been evaluated in order to add to the knowledge of digital forensic investigators. Lastly, the budgeting recommendations for forensic investigators have also been explained, concluding that the budgeting model is suitable to be used for time estimation for password cracking cases. The next chapter consists of the conclusion of the thesis.

Chapter 6

Conclusion

6.0 RESEARCH SUMMARY

The importance of passwords along with their advantages and disadvantages were explored in Chapter 1. Passwords can be used for good in data security; however, passwords can also be used by malicious people to secure incriminating information. There are many challenges for the forensic investigator in the field of password cracking. These challenges, together with past research performed in the area, were motivational factors for performing this research.

The research study was initiated in Chapter 2 by performing a literature review of the relevant theory in the field of password cracking. For the purpose of this research, password schemes in operating systems such as Linux were reviewed. The various password-recovery options for the forensic investigator and the existing password-recovery strategies using Password Recovery Toolkit and Distributed Network Attack were also reviewed. Password-cracking tools and techniques were also identified. Various problems related to costing and password cracking were explained, and the miscellaneous past research that helped influence this research was tabulated. Lastly, the key problems and issues in the area of password cracking were also identified.

In Chapter 3, the research methodology was developed by reviewing the similar past research studies performed in the area. This review of similar studies, along with the literature review, formed the basis of the development of the research design. The research design included the budgeting model as well as the experimental design. The research questions and hypotheses and also the various phases of research were also identified. The data specifications required for the research were also defined.

After the experiment was performed, the fieldwork performed was reported in Chapter 4. The deviations from the specifications were identified. The data collection performed along with the data analysis and the presentation findings were also presented.

Chapter 5 discussed the findings. The research questions were discussed and answered based on the results. The research hypotheses were also tested in order to determine whether they were true or false. The research findings were also discussed with relation to the theory and methodology defined in Chapters 2 and 3. The entire research design and process were evaluated and discussed in order to gain and add to existing best-practice knowledge. Budgeting recommendations for forensic investigators were presented, based on what was learned from this research study.

Finally, this chapter concludes the research thesis by presenting the key findings of the research. Section 6.1 reviews the summary of findings. Section 6.2 provides the summary of the answer to all the research questions. The conclusion and areas of future research are discussed in section 6.3.

6.1 SUMMARY OF FINDINGS

The data collection and research findings have been performed as shown in Chapter 4. The fieldwork performed and also the necessary deviations from the research plan have been discussed. This section provides a summary of the findings that have been presented in Chapter 4.

For the experiment, the various time budgets for the blocks of passwords were calculated using the budgeting model. The time budgets calculated before the experiment are shown in Table 6.1.

Table 6.1: Time budgets for all the blocks

Block Number	**Budgeted Time**
1	10 minutes
2	2 hours 56 minutes
3	22 hours 42 minutes
4	24.8 days
TOTAL AVAILABLE TIME BUDGET:	**7 Days**

As shown in Table 6.1, the time budgets for each block or group of 50 passwords were calculated. The overall budget for the entire experiment was seven days, based on the practical limitations of time and resources. After calculating the time budgets, the passwords were added to Access Data's Distributed Network Attack password cracking suite, which was used to crack the passwords. After the password cracking procedure, the password cracking reports were generated using Distributed Network Attack. The actual times taken to crack the passwords were considered for analysis with the aid of Microsoft Excel.

The results of the individual blocks are shown in Appendices 3 and 4. The data findings of the individual blocks are also presented in Section 4.4.1. As a result of the experiment, all of the passwords were cracked within the allocated time budgets. However, Block 4 was incomplete, since it was not possible to leave it to crack for its allocated time budget due to the constraints of the overall time budget available. The individual accounts in each of the blocks had variable password cracking times. The comparison of the actual time taken to crack the entire blocks of passwords with the budgeted time is shown in Table 6.2.

Table 6.2: The actual time and the budgeted time for each block

Block Number	**Actual Time (in Seconds)**	**Budgeted Time (in Seconds)**
1	210	557.2
2	6749	10556.91
3	42710	81681

As shown in Table 6.2, the actual time in seconds for block 1 was 210 seconds while the budgeted time was 557.2 seconds. Also, the actual times for blocks 2 and 3 were 6,749 and 42,710 seconds, while their budgeted times were 10,556.91 and 81,681 seconds respectively.

Therefore, as per the findings, the individual accounts in each of the blocks had variable password cracking times. Also, all of the passwords in blocks 1, 2, and 3 were cracked within the times allocated by the budgeting model. Therefore, the findings demonstrate that the times taken to crack the passwords were less than time allocated by the budgeting model.

The budgeting model and the research performed help in adding to best-practice knowledge for digital forensic investigators. The next section will summarise the answers to the research questions.

6.2 ANSWERS TO RESEARCH QUESTIONS

Based on the findings presented in Chapter 4, Chapter 5 answered the research questions and tested the hypotheses defined in Chapter 3. This section summarises all the answers to the research questions along with the results of the hypothesis tests.

A summary of all of the research questions defined in Section 3.2.4 and the answers found in Section 5.1 is shown in Table 6.3 below.

Table 6.3: Research questions and the respective research answers

RESEARCH QUESTION	**ANSWER**
MAIN RESEARCH QUESTION*:* *What are the time implications of cracking passwords using the budgeting model?"*	*A: Based on the research findings, the times allocated to the blocks of passwords depict an inconclusive result. The findings also signify that the actual times required to crack the passwords crack are very near to half of the existing allocated time budgets.*
SECONDARY RESEARCH QUESTION 1: *How many passwords can be cracked within the allocated time budget?*	*A1: For the current experiment, all of the 50 passwords in blocks 1, 2 and 3 were cracked successfully within the allocated budget. Therefore, based on the results of the experiment, it has been demonstrated that all of the given passwords can be cracked within the time budget allocated using the budgeting model.*
SECONDARY RESEARCH QUESTION 2: *How much time is required to crack all of the passwords in each of the given blocks?*	*A2: Based on the results of this study, the amount of time required to crack all of the passwords in blocks 1, 2, and 3 were 210, 6,749 and 42,710 seconds, respectively. The actual amount of time required to crack all of the passwords in block 4 is unknown.*

SECONDARY RESEARCH QUESTION 3: *What are the guidelines for best practice advice for digital forensic investigators in the field of cracking passwords using AccessData's Distributed Network Attack (DNA)?*	*A3: Based on the results of the study, all of the passwords were cracked within the given budget. Thus, the budgeting model was successfully implemented and operationalised within its given constraints and limitations. The hypothesis that 'All of the passwords can be cracked within the time budget allocated by the use of the budgeting model' has also been proved to be true. Therefore, the budgeting model guidelines are suitable to be followed as best-practice advice for password cracking using AccessData's Distributed Network Attack. The constraints and limitations of the budgeting model must also be considered and it should be used in conjunction with the existing password-recovery strategies and best practices.*

The main research question related to the time implications of cracking passwords using the budgeting model was defined in Section 3.2.1. It was discovered, based on the research findings in Chapter 4 that the times required to crack the password were less than the times allocated by the budgeting model. Another important finding based on the results was that the actual times required to crack the passwords were very near to half of the budgeted time. The secondary research questions are also answered as shown above in Table 6.3.

The research hypotheses defined in Section 3.2.4 were based on the evaluation of the budgeting model proposed in Section 3.2.1. The budgeting model was evaluated based on the results of the experiment (see Section 5.1.2). The hypotheses tested and the results of the tests are summarised in Table 6.4 below.

Table 6.4: Hypothesis test results

Hypothesis Tested	**Result**
Hypothesis H0: *All of the passwords can be cracked within the time budget allocated by the use of the budgeting model.*	Accepted

Hypothesis H1: *None of the passwords can be cracked within the time budget allocated by the use of the budgeting model.*	Rejected
Hypothesis H2: *Some but not all of the passwords can be cracked within the time budget allocated by the use of the budgeting model.*	Rejected

As shown in Table 6.4, out of the three hypotheses tested, only H0 was accepted, whereas H1 and H2 were rejected. Therefore, based on the accepted hypothesis, all of the passwords can be cracked in less than the time budget allocated by the budgeting model.

Thus, the research found that the budgeting model can be successfully implemented. It can be said that, by using the budgeting model within its defined constraints and limitations, all of the passwords can be cracked in less than their allocated times. It can also be said that, on an average, the passwords were cracked in half of the allocated time. Based on the research study, the recommended budgeting procedure for forensic investigators has been presented in Section 5.3.

6.3 CONCLUSION AND FUTURE RESEARCH

This chapter concludes the research thesis. The research performed has proposed a budgeting model and evaluated it by simulating various hypothetical password cracking cases. The research study performed has also helped in increasing understanding of the processes involved in cracking passwords. Therefore, the research has helped in adding to the knowledge of best practices and processes involved in order to crack a password. This study has also proved demonstrated that the budgeting model is suitable to be used by digital forensic investigators in real-life scenarios. If the budgeting model is used within its constraints and limitations, digital forensic investigators would on an average find the password cracking times to be half of the budgeted time.

There are many problem areas that require future research. The problem faced by digital forensic investigators in the field of password cracking is due to the conflict between security and forensics. Due to increasing security, digital forensic investigators and law enforcement agencies face various challenges to overcome the security barriers of persons with malicious and criminal intent. This research also had its limitations and constraints. From a range of problems in the field of password cracking, this research selected one problem and provided a possible solution. This research also proposed what a budgeting model may be like. The limitations of the budgeting model and the processes used in the research may not necessarily be generalised to every context, case, or system. This research was also a simulation of various hypothetical cases. Therefore, based on the research performed, the various areas for further research involve the use of the budgeting model and procedures for studying real-life case scenarios.

Further areas of research include improvements in the underlying password -cracking technology. Improvements in password cracking technologies would improve the speed of password cracking and thus also reduce the costs. These areas of further research could also be combined and further research could also be done to improve the underlying technology along with the processes followed by the digital forensic investigator in order to crack passwords in a cost-effective manner. The budgeting model and its processes can also be further researched and improved by performing studies on real-life case scenarios.

References

AccessData Corp. (2006). *Password Recovery with PRTK / DNA.* U.S.A.: Author.

AccessData Corp. (2010). *User Guide: Password Recovery Toolkit.* U.S.A.: Author.

Atrill, P., McLaney, E., Harvey, D., & Jenner, M. (2006). *Accounting: an Introduction (3rd Ed.)*. Australia: Pearson Education.

Bengtsson, J. (2007). Parallel Password Cracker: A Feasibility Study of Using Linux Clustering Technique in Computer Forensics. In *Second International Workshop on Digital Forensics and Incident Analysis (WDFIA 2007)*. (pp. 75-82). Karlovassi, Samos, Greece: IEEE.

Berger, Y., Wool, A., & Yeredor, A. (2006). Dictionary Attacks Using Keyboard Acoustic Emanations. In *Proceedings of the 13th ACM conference on Computer and communications security* (pp. 245-254). Alexandria, Virginia, USA: ACM.

Britz, M. T. (2009). *Computer Forensics and Cyber Crime.* New Jersey, U.S.A: Prentice Hall.

Burnett, M., & Kleiman, D. (2006). *Perfect passwords: selection, protection, authentication.* Canada: Syngress.

Casey, E. (2004). *Digital evidence and computer crime: forensic science, computers and the Internet – (2nd Ed).* London, UK: Academic Press.

Casey, E. (2006). Cutting corners: Trading justice for cost savings. *Digital investigation, 3*(4), 185 – 186.

Cisneros, R., Bliss, D., & Garcia, M. (2006). Password Auditing Applications. *Journal of Computing Sciences in Colleges, 21*(4), 196-202.

Clayton, R., & Bond, M. (2003). Experience Using a Low-Cost FPGA Design to Crack DES Keys. *Lecture Notes in Computer Science, 2523/2003*(1), 877-883.

Craiger, J.,P., Swauger, J., & Marberry, C. (2005). Digital evidence obfuscation: recovery techniques. In *Proceedings of the Society for Optical Engineering Conference* (pp. 587-594). Orlando, FL, U.S.A: SPIE.

Dandass, Y., S. (2008). Using FPGAs to Parallelize Dictionary Attacks for Password Cracking. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences* (pp. 486-493). Waikoloa, Big Island, Hawaii: IEEE Computer Society.

Dell'Amico, M., Michiardi, P., & Roudier, Y. (2010). Password Strength: An Empirical Analysis. In *Proceedings of IEEE INFOCOM 2010*.(pp. 1-9). San Diego, CA, USA: IEEE.

Dube, D., P., & Gulati, V., P. (2005). *Information System Audit and Assurance.* New Delhi, India: Tata McGraw-Hill.

Frichot, C. (2004). *An Analysis and Comparison of Clustered Password Crackers.* Paper presented at the 2nd Australian Computer, Network and Information Conference, Western Australia.

Frykholm, N., & Juels, A. (2001). Error-tolerant password recovery. In *Proceedings of the 8th ACM conference on Computer and Communications Security* (pp. 1-9). Philadelphia, PA, USA: ACM.

Garfinkel, S., Spafford, G., & Schwartz, A. (2003). *Practical Unix and Internet Security*. USA.: O'Reilly.

Graves, R., E. (2008). *High performance password cracking by implementing rainbow tables on nVidia graphics cards (IseCrack).* Masters Thesis, Iowa State University, Ames, Iowa, U.S.A. Retrieved October 15, 2009, from http://gradworks.umi.com.ezproxy.aut.ac.nz/14/61/1461850.html

Klein, D. (1990). ‘‘Foiling the Cracker’’: A Survey of, and Improvements to, Password Security. In *Proceedings of the 2nd USENIX Security Workshop (pp. 1 – 11).* Portland, Oregon, USA.

Kleinhans, H., Butts, J., & Shenoi, S. (2009). Password Cracking using Sony Playstations. In P. Gilbert & S. Shenoi (Eds.), *Advances in Digital Forensics V, IFIP AICT 306* (pp. 215-227). Boston: Springer.

L0pht Holdings. (2009). *L0phtCrack Password Auditor v6 – Documentation.* Retrieved from http://www.l0phtcrack.com/help/index.html

Lee, S., Savoldi, A., Lee, S., & Lim, J. (2007). Password Recovery Using an Evidence Collection Tool and Countermeasures. In *Third International Conference on International Information Hiding and Multimedia Signal Processing 2007* (pp. 97-102). Kaohsiung, Taiwan: IEEE Computer Society.

Lucey, T. (2002). *Costing* (6th ed.). China: Thomson Learning.

Marechal, S. (2008). Advances in password cracking. *Journal in computer virology, 4*(1), 73 – 81.

McClure, S., Scambray, J. & Kurtz, G. (2009). *Hacking Exposed 6: Network Security secrets & solutions.* U.S.A.: McGraw Hill.

Mentens, N., Batina, L., Preneel, B., & Verbauwhede, I. (2005). Cracking Unix Passwords using FPGA Platforms. In *SHARCS’05* (pp. 1-9). Paris, France.

Mookhey, K., K. (2004). Open Source Tools for Security and Control Assessment. *Information Systems Control Journal, 1*(1), 39-44.

Oechslin, P. (2003). Making a Faster Cryptanalytic Time-Memory Trade-Off. *Lecture Notes in Computer Science, 2729/2003*(1), 617-630.

Openwall Project. (2010). *John the Ripper documentation.* Retrieved from http://www.openwall.com/john/doc/

Peikari, C., & Chuvakin, A. (2004). *Security Warrior.* U.S.A.: O'Reilly.

Potter, B. (2009). A review of L0phtCrack 6. *Network security, 2009*(7), 14-17.

Raval, V., & Fichadia, A. (2007). *Risks, Controls, and security : Concepts and Applications*. Hoboken, NJ : Wiley.

Rowan, T. (2009). Password protection: the next generation. *Network security, 1*(2), 4 – 7.

Salomon, D. (2006). *Foundations of Computer Security.* U.S.A.: Springer.

Shinder, D., L., & Tittel, E. (2002). *Scene of the cybercrime: computer forensics handbook.* U.S.A: Syngress.

Skoudis, E. (2007). A new breed of hacker tools and defenses. In H. F. Tipton & M. Krause (Eds.), *Information Security Management Handbook* (6th ed., pp. 951-964). U.S.A. : Auerbach Publications.

Stallings, W.(2006). *Cryptography and Network Security: Principles and Practices (4th Ed.).*USA: Prentice Hall.

Thing, V.,L.,L., & Ying, H-M. (2009). A novel time-memory trade-off method for password recovery. *Digital investigation, 6*(1), S114-S120.

Toxen, B. (2003). *Real world Linux security: intrusion, prevention, detection and recovery.* New Jersey, U.S.A.: Pearson Education, Inc.

Viega, J., & Messier, M. (2003). *Secure programming cookbook for C and C++.*U.S.A.: O'Reilly.

Weir, M., Aggarwal, S., Medeiros, B., D., & Glodek, B. (2009). Password Cracking Using Probabilistic Context-Free Grammars. In *2009 30th IEEE Symposium on Security and Privacy* (pp. 391-405). Oakland, California, U.S.A: IEEE Computer Society.

Wiles, J., & Reyes, A. (2007). *Best Damn Cybercrime and Digital Forensics Book Period.* U.S.A.: Syngress.

APPENDIX 1: Time Budget Calculation of All Blocks

The Table below shows AccessData's rules (or population sets used), along with the size of the population, password chosen, and respective time budgets calculated.

Sno	ID	Desc	Population	Password	Time Required in Seconds
		BLOCK#1			
1	Bas -1-01	One digit search	10	9	0.0001000000
2	Bas -1-02	One letter, language specific search	52	O	0.0005200000
3	Bas -1-03	Two digit search	100	94	0.0010000000
4	Adv -1-01	All one-character, language specific search	256	?	0.0025600000
5	Bas -1-05	Three Digit search	1,000	173	0.0100000000
6	Bas -1-04	Two letter, language specific search	2,704	gS	0.0270400000
7	Bas -1-07	Four digit search	10,000	3482	0.1000000000
8	Bas -2-17	Dictionary primary search ([EN-1] Common-en-c.adf)	22,708	privs	0.2270800000
9	Bas -2-18	Dictionary primary reverse search ([EN-1] Common-en-c.adf)	22,708	trauts	0.2270800000
10	Adv -1-02	All two character, language specific search	65,536	U%	0.6553600000
11	Bas -1-08	Five digit search	100,000	91426	1.0000000000
12	Bas -2-19	Dictionary with two characters uppercased search <([EN-1] Common-en-c.adf)>	103,065	sOcKs	1.0306500000
13	Bas -2-17	Dictionary primary search ([EN-2] Miscellaneous-en-c.adf)	109,840	chicle	1.0984000000
14	Bas -2-18	Dictionary primary reverse search <([EN-2] Miscellaneous-en-c.adf)>	109,840	edimanoflus	1.0984000000
15	Bas -1-06	Three letter, language specific search	140,608	aHv	1.4060800000
16	PP 1-03	Dictionary preceded by a verb or prepositional phrase search ([EN-1] Common-en-c.adf)	141,925	ofasango	1.4192500000
17	Bas -2-20	Dictionary primary character replacements search ([EN-1] Common-en-c.adf)	175,020	m3r(ury	1.7502000000
18	Bas -2-23	Dictionary primary followed by a one digit search <([EN-1] Common-en-c.adf)>	227,080	practice7	2.2708000000
19	Bas -2-24	Dictionary primary preceded by a one digit search ([EN-1] Common-en-c.adf)	227,080	8holly	2.2708000000
20	PP 1-04	The common english dictionary preceded by a verb or prepositional phrase search ([EN-1] Common-en-c.adf)	283,850	goingtotheolivier	2.8385000000
21	Bas -2-21	Dictionary primary followed by common postfixes search <([EN-1] Common-en-c.adf)>	590,408	joseness	5.9040800000
22	Bas -2-19	Dictionary with two characters uppercased search ([EN-2] Miscellaneous-en-c.adf)	591,894	8uCE	5.9189400000
23	PP 1-03	Dictionary preceded by a verb or prepositional phrase search ([EN-2] Miscellaneous-en-c.adf)	686,500	intothe5oid	6.8650000000
24	Bas -2-22	Dictionary primary preceded by common prefixes search ([EN-1] Common-en-c.adf)	772,072	outroy	7.7207200000

25	Bas -2-27	Dictionary primary followed by a non-alphanumeric symbol search <([EN-1] Common-en-c.adf)>	794,780	salmon$	7.9478000000
26	Bas -2-28	Dictionary primary preceded by a language-specific non-alphanumeric symbol search ([EN-1] Common-en-c.adf)	794,780	~n2netsurfn	7.9478000000
27	Bas -2-20	Dictionary primary character replacements search <([EN-2] Miscellaneous-en-c.adf)>	869,604	6l1stic	8.6960400000
28	Bas -1-10	Six digit search	1,000,000	517642	10.0000000000
29	Bas -2-23	Dictionary primary followed by a one digit search ([EN-2] Miscellaneous-en-c.adf)	1,098,400	5xt2	10.9840000000
30	Bas -2-24	Dictionary primary preceded by a one digit search <([EN-2] Miscellaneous-en-c.adf)>	1,098,400	86read	10.9840000000
31	Bas -2-25	Dictionary primary followed by a one letter, language specific search <([EN-1] Common-en-c.adf)>	1,180,816	sneakyb	11.8081600000
32	Bas -2-26	Dictionary primary preceded by a one letter language specific search ([EN-1] Common-en-c.adf)	1,180,816	xrate	11.8081600000
33	Bas -2-17	Dictionary primary search ([EN-4] General-1-en-c.adf)	1,342,860	foreconceive	13.4286000000
34	Bas -2-17	Dictionary primary search ([EN-4] General-2-en-c.adf)	1,342,860	regla	13.4286000000
35	Bas -2-18	Dictionary primary reverse search <([EN-4] General-1-en-c.adf)>	1,342,860	kcirttah	13.4286000000
36	Bas -2-18	Dictionary primary reverse search ([EN-4] General-2-en-c.adf)	1,342,860	esnenaoros	13.4286000000
37	Bas -2-17	Dictionary primary search ([EN-3] Names-en-c.adf)	1,493,412	maung	14.9341200000
38	Bas -2-18	Dictionary primary reverse search ([EN-3] Names-en-c.adf)	1,493,412	asjak	14.9341200000
39	Bas -2-29	Dictionary primary character replacement , followed by a one digit search <([EN-1] Common-en-c.adf)>	1,750,200	m0th3r8	17.5020000000
40	Bas -2-30	Dictionary primary character replacement, preceded by a one digit search <([EN-1] Common-en-c.adf)>	1,750,200	10ff	17.5020000000
41	Bas -2-31	Dictionary primary preceded and followed by a one digit search <([EN-1] Common-en-c.adf)>	2,270,800	9tea7	22.7080000000
42	Bas -2-32	Dictionary primary followed by a two digits search ([EN-1] Common-en-c.adf)	2,270,800	sensor56	22.7080000000
43	Bas -2-33	Dictionary primary preceded by a two digits search ([EN-1] Common-en-c.adf)	2,270,800	30pookie	22.7080000000
44	Adv -1-07	One language specific character followed by a four digit search	2,560,000	X9825	25.6000000000
45	Bas -2-21	Dictionary primary followed by common postfixes search ([EN-2] Miscellaneous-en-c.adf)	2,855,840	5sidenceites	28.5584000000
46	Bas -2-37	Three letter, language specific characters followed by common postfixes	3,655,808	xYbed	36.5580800000
47	Bas -2-22	Dictionary primary preceded by common prefixes search <([EN-2] Miscellaneous-en-c.adf)>	3,734,560	macasylums	37.3456000000
48	Bas -2-27	Dictionary primary followed by a non-alphanumeric symbol search ([EN-2] Miscellaneous-en-c.adf)	3,844,400	4vies&	38.4440000000
49	Bas -2-28	Dictionary primary preceded by a language-specific non-alphanumeric symbol search <([EN-2] Miscellaneous-en-c.adf)>	3,844,400	$5sterone	38.4440000000
50	Bas -3-01	Dictionary primary with a non-alphanumeric symbol inserted search ([EN-1] Common-en-c.adf)	4,152,120	pyr%amid	41.5212000000

			TOTAL TIME REQUIRED FOR WHOLE BLOCK 1 :		10 minutes
		BLOCK#2			
51	Bas -2-38	Three letter, language specific characters preceded by common prefixes	4,780,672	conpaL	47.8067200000
52	Bas -2-36	Date search (2 digit year)	4,840,000	20april89	48.4000000000
53	Bas -2-25	Dictionary primary followed by a one letter, language specific search ([EN-2] Miscellaneous-en-c.adf)	5,711,680	4wrightb	57.1168000000
54	Bas -2-26	Dictionary primary preceded by a one letter language specific search <([EN-2] Miscellaneous-en-c.adf)>	5,711,680	znetcom	57.1168000000
55	Bas -2-35	Dictionary primary preceded by one digit followed by common postfixes <([EN-1] Common-en-c.adf)>	5,904,080	5missioners	59.0408000000
56	Bas -2-01	Four letter, language specific search	7,311,616	tAbU	73.1161600000
57	Bas -2-34	Dictionary primary preceded by common prefixes and followed by a one digit search <([EN-1] Common-en-c.adf)>	7,720,720	nonnaturfot9	77.2072000000
58	PP 1-03	Dictionary preceded by a verb or prepositional phrase search ([EN-4] General-1-en-c.adf)	8,392,875	wouldbeemotionalization	83.9287500000
59	PP 1-03	Dictionary preceded by a verb or prepositional phrase search ([EN-4] General-2-en-c.adf)	8,392,875	isintheslath	83.9287500000
60	Bas -2-29	Dictionary primary character replacement , followed by a one digit search <([EN-2] Miscellaneous-en-c.adf)>	8,696,040	61i0n8	86.9604000000
61	Bas -2-30	Dictionary primary character replacement, preceded by a one digit search <([EN-2] Miscellaneous-en-c.adf)>	8,696,040	15o1nt	86.9604000000
62	PP 1-03	Dictionary preceded by a verb or prepositional phrase search ([EN-3] Names-en-c.adf)	9,333,825	couldolliges	93.3382500000
63	Bas -2-08	Seven digit search	10,000,000	5945259	100.0000000000
64	Bas -2-19	Dictionary with two characters uppercased search ([EN-3] Names-en-c.adf)	10,369,430	heNNigar	103.6943000000
65	Bas -2-31	Dictionary primary preceded and followed by a one digit search <([EN-2] Miscellaneous-en-c.adf)>	10,984,000	05usite3	109.8400000000
66	Bas -2-32	Dictionary primary followed by a two digits search ([EN-2] Miscellaneous-en-c.adf)	10,984,000	6ulate27	109.8400000000
67	Bas -2-33	Dictionary primary preceded by a two digits search <([EN-2] Miscellaneous-en-c.adf)>	10,984,000	48illegal	109.8400000000
68	Bas -2-23	Dictionary primary followed by a one digit search ([EN-4] General-1-en-c.adf)	13,428,600	gelosin8	134.2860000000
69	Bas -2-23	Dictionary primary followed by a one digit search ([EN-4] General-2-en-c.adf)	13,428,600	undercapitalised3	134.2860000000
70	Bas -2-24	Dictionary primary preceded by a one digit search ([EN-4] General-1-en-c.adf)	13,428,600	7hagiographic	134.2860000000
71	Bas -2-24	Dictionary primary preceded by a one digit search <([EN-4] General-2-en-c.adf)>	13,428,600	6unoffendable	134.2860000000
72	Bas -2-20	Dictionary primary character replacements search ([EN-3] Names-en-c.adf)	13,761,144	s4@ittariid	137.6114400000

73	Bas -2-20	Dictionary primary character replacement search ([EN-4] General-2-en-c.adf)	14,342,144	th3na	143.4214400000
74	Bas -2-19	Dictionary with two characters uppercased search ([EN-4] General-1-en-c.adf)	14,631,887	ideoLaTry	146.3188700000
75	Bas -2-20	Dictionary primary character replacements search <([EN-4] General-1-en-c.adf)>	14,818,484	gurd1n@	148.1848400000
76	Bas -2-23	Dictionary primary followed by a one digit search ([EN-3] Names-en-c.adf)	14,934,120	schleifer5	149.3412000000
77	Bas -2-24	Dictionary primary preceded by a one digit search ([EN-3] Names-en-c.adf)	14,934,120	4monro	149.3412000000
78	Bas -2-19	Dictionary with two characters uppercased search ([EN-4] General-2-en-c.adf)	15,995,902	slateYaRd	159.9590200000
79	Adv -1-03	All three character, language specific search	16,777,216	c&G	167.7721600000
80	Bas -3-02	Dictionary primary character replacement, followed by a two digit search <([EN-1] Common-en-c.adf)>	17,502,000	z0mb1e39	175.0200000000
81	Bas -3-03	Dictionary primary character replacement, preceded by a two digit search <([EN-1] Common-en-c.adf)>	17,502,000	92ars3nal	175.0200000000
82	Bas -2-12	7- digit telephone number search	20,000,000	9642861	200.0000000000
83	Bas -3-01	Dictionary primary with a non-alphanumeric symbol inserted search ([EN-2] Miscellaneous-en-c.adf)	21,170,380	eth#ics	211.7038000000
84	Bas -3-04	Dictionary primary followed by a three digit search ([EN-1] Common-en-c.adf)	22,708,000	key395	227.0800000000
85	Bas -3-05	Dictionary primary preceded by a three digit search ([EN-1] Common-en-c.adf)	22,708,000	832nail	227.0800000000
86	Bas -2-35	Dictionary primary preceded by one digit followed by common postfixes <([EN-2] Miscellaneous-en-c.adf)>	28,558,400	5disadvantage abc	285.5840000000
87	PP 1-01	Two word concatenation without spaces search ([EN-1] Common-en-c.adf)	32,228,329	mechaniccorra do	322.2832900000
88	PP 1-02	Two word concatenation with spaces search <([EN-1] Common-en-c.adf)>	32,228,329	oranges tie	322.2832900000
89	Bas -2-21	Dictionary primary followed by common postfixes search ([EN-4] General-1-en-c.adf)	34,914,360	gubioa2b	349.1436000000
90	Bas -2-21	Dictionary primary followed by common postfixes search ([EN-4] General-2-en-c.adf)	34,914,360	subjing	349.1436000000
91	Bas -2-34	Dictionary primary preceded by common prefixes and followed by a one digit search <([EN-2] Miscellaneous-en-c.adf)>	37,345,600	bio5tient	373.4560000000
92	Bas -2-21	Dictionary primary followed by common postfixes search ([EN-3] Names-en-c.adf)	38,828,712	nauenbergers	388.2871200000
93	Bas -2-22	Dictionary primary preceded by common prefixes search <([EN-4] General-1-en-c.adf)>	45,657,240	posthattr	456.5724000000
94	Bas -2-22	Dictionary primary preceded by common prefixes search ([EN-4] General-2-en-c.adf)	45,657,240	disstomps	456.5724000000
95	Bas -2-27	Dictionary primary followed by a non-alphanumeric symbol search ([EN-4] General-1-en-c.adf)	47,000,100	kitsipki^	470.0010000000

96	Bas -2-27	Dictionary primary followed by a non-alphanumeric symbol search ([EN-4] General-2-en-c.adf)	47,000,100	trapezoidal@	470.0010000000
97	Bas -2-28	Dictionary primary preceded by a language-specific non-alphanumeric symbol search <([EN-4] General-1-en-c.adf)>	47,000,100	:frontopolar	470.0010000000
98	Bas -2-28	Dictionary primary preceded by a language-specific non-alphanumeric symbol search ([EN-4] General-2-en-c.adf)	47,000,100	|syndesis	470.0010000000
99	Bas -2-22	Dictionary primary preceded by common prefixes search ([EN-3] Names-en-c.adf)	50,776,008	antikillone	507.7600800000
100	Bas -2-27	Dictionary primary followed by a non-alphanumeric symbol search ([EN-3] Names-en-c.adf)	52,269,420	naufal	522.6942000000
			TOTAL TIME REQUIRED FOR WHOLE BLOCK 2 :		2 hours 56 minutes
		BLOCK#3			
101	Bas -2-28	Dictionary primary preceded by a language-specific non-alphanumeric symbol search ([EN-3] Names-en-c.adf)	52,269,420	<lemayne	522.6942000000
102	Adv -1-22	Dictionary primary preceded by a two digit followed by common postfixes <([EN-1] Common-en-c.adf)>	59,040,800	57isobared	590.4080000000
103	Adv -1-20	Dictionary primary followed by a two letter, language specific search ([EN-1] Common-en-c.adf)	61,402,432	individualbc	614.0243200000
104	Adv -1-21	Dictionary primary preceded by a two letter, language specific search ([EN-1] Common-en-c.adf)	61,402,432	pretendyu	614.0243200000
105	PP 2-03	Two word passphrase using the common english dictionary ([EN-1] Common-en-c.adf)	64,456,658	lasernight	644.5665800000
106	Adv -1-09	Two language-specific characters followed by a three digit search	65,536,000	g#593	655.3600000000
107	Bas -2-25	Dictionary primary followed by a one letter, language specific search ([EN-4] General-1-en-c.adf)	69,828,720	creuxu	698.2872000000
108	Bas -2-25	Dictionary primary followed by a one letter, language specific search ([EN-4] General-2-en-c.adf)	69,828,720	rioritya	698.2872000000
109	Bas -2-26	Dictionary primary preceded by a one letter language specific search <([EN-4] General-1-en-c.adf)>	69,828,720	venprisonen	698.2872000000
110	Bas -2-26	Dictionary primary preceded by a one letter language specific search ([EN-4] General-2-en-c.adf)	69,828,720	gseech	698.2872000000
111	Adv -1-23	Dictionary primary preceded by common prefixes and followed by a two digit search <([EN-1] Common-en-c.adf)>	77,207,200	projewels96	772.0720000000
112	Bas -2-25	Dictionary primary followed by a one letter, language specific search ([EN-3] Names-en-c.adf)	77,657,424	oversonB	776.5742400000
113	Bas -2-26	Dictionary primary preceded by a one letter language specific search ([EN-3] Names-en-c.adf)	77,657,424	Jcicek	776.5742400000
114	Bas -3-02	Dictionary primary character replacement, followed by a two digit search <([EN-2] Miscellaneous-en-c.adf)>	86,960,400	p4dr334	869.6040000000
115	Bas -3-03	Dictionary primary character replacement, preceded by a two digit search <([EN-2] Miscellaneous-en-c.adf)>	86,960,400	055men4	869.6040000000

116	Bas -2-13	Eight Digit Search	100,000,000	67298157	1000.0000000000
117	Bas -3-04	Dictionary primary followed by a three digit search ([EN-2] Miscellaneous-en-c.adf)	109,840,000	5rvants167	1098.4000000000
118	Bas -3-05	Dictionary primary preceded by a three digit search <([EN-2] Miscellaneous-en-c.adf)>	109,840,000	6277polis	1098.4000000000
119	Bas -2-31	Dictionary primary preceded and followed by a one digit search <([EN-4] General-1-en-c.adf)>	134,286,000	7galvanometer s9	1342.8600000000
120	Bas -2-31	Dictionary primary preceded and followed by a one digit search <([EN-4] General-2-en-c.adf)>	134,286,000	2surle4	1342.8600000000
121	Bas -2-32	Dictionary primary followed by a two digits search ([EN-4] General-1-en-c.adf)	134,286,000	floorings73	1342.8600000000
122	Bas -2-32	Dictionary primary followed by a two digits search ([EN-4] General-2-en-c.adf)	134,286,000	ricochet24	1342.8600000000
123	Bas -2-33	Dictionary primary preceded by a two digits search <([EN-4] General-1-en-c.adf)>	134,286,000	emaciating48	1342.8600000000
124	Bas -2-33	Dictionary primary preceded by a two digits search ([EN-4] General-2-en-c.adf)	134,286,000	64pseud	1342.8600000000
125	Bas -2-29	Dictionary primary character replacement , followed by a one digit search <([EN-3] Names-en-c.adf)>	137,611,440	m4ll4m5	1376.1144000000
126	Bas -2-30	Dictionary primary character replacement, preceded by a one digit search <([EN-3] Names-en-c.adf)>	137,611,440	31ndr4y4n	1376.1144000000
127	Bas -2-29	Dictionary primary character replacement , followed by a one digit search <([EN-4] General-2-en-c.adf)>	143,421,440	r3gl0ve5	1434.2144000000
128	Bas -2-30	Dictionary primary character replacement, preceded by a one digit search <([EN-4] General-2-en-c.adf)>	143,421,440	8str1dul4	1434.2144000000
129	Bas -2-29	Dictionary primary character replacement , followed by a one digit search <([EN-4] General-1-en-c.adf)>	148,184,840	dr4@@er$0	1481.8484000000
130	Bas -2-30	Dictionary primary character replacement, preceded by a one digit search <([EN-4] General-1-en-c.adf)>	148,184,840	13ffortfu1	1481.8484000000
131	Bas -2-31	Dictionary primary preceded and followed by a one digit search <([EN-3] Names-en-c.adf)>	149,341,200	5ogawara3	1493.4120000000
132	Bas -2-32	Dictionary primary followed by a two digits search ([EN-3] Names-en-c.adf)	149,341,200	nkwazo92	1493.4120000000
133	Bas -2-33	Dictionary primary preceded by a two digits search ([EN-3] Names-en-c.adf)	149,341,200	20niegel	1493.4120000000
134	Adv -1-05	One digit followed by three language specific characters	167,772,160	4i(D	1677.7216000000
135	Adv -1-06	Three language specific characters followed by one digit search	167,772,160	{Y^8	1677.7216000000
136	PP 2-01	Word inserted into another word search ([EN-1] Common-en-c.adf)	168,368,466	lyryannn	1683.6846600000
137	Bas -3-06	Four letter, language specific characters followed by common postfixes	190,102,016	udcj!@#	1901.0201600000
138	Adv -1-24	Dictionary primary preceded and followed by a two digit search <([EN-1] Common-en-c.adf)>	227,080,000	89quick24	2270.8000000000
139	Adv -1-25	Dictionary primary followed by a four digit search ([EN-1] Common-en-c.adf)	227,080,000	tryscer2374	2270.8000000000

140	Adv -1-26	Dictionary primary preceded by a four digit search ([EN-1] Common-en-c.adf)	227,080,000	1943percolate	2270.8000000000
141	Bas -3-07	Four letter, language specific characters preceded by common prefixes	248,594,944	non-fyhy	2485.9494400000
142	Adv -1-22	Dictionary primary preceded by a two digit followed by common postfixes <([EN-2] Miscellaneous-en-c.adf)>	285,584,000	34deadensing	2855.8400000000
143	Adv -1-20	Dictionary primary followed by a two letter, language specific search ([EN-2] Miscellaneous-en-c.adf)	297,007,360	5etudeje	2970.0736000000
144	Adv -1-21	Dictionary primary preceded by a two letter, language specific search <([EN-2] Miscellaneous-en-c.adf)>	297,007,360	asfaculty	2970.0736000000
145	Bas -3-01	Dictionary primary with a non-alphanumeric symbol inserted search ([EN-3] Names-en-c.adf)	342,745,200	oshi$ka	3427.4520000000
146	Bas -2-35	Dictionary primary preceded by one digit followed by common postfixes <([EN-4] General-1-en-c.adf)>	349,143,600	2exsteties	3491.4360000000
147	Bas -2-35	Dictionary primary preceded by one digit followed by common postfixes <([EN-4] General-2-en-c.adf)>	349,143,600	5snapbackers	3491.4360000000
148	Adv -1-23	Dictionary primary preceded by common prefixes and followed by a two digit search <([EN-2] Miscellaneous-en-c.adf)>	373,456,000	outatalanta	3734.5600000000
149	Bas -2-02	Five letter, language specific search	380,204,032	bDtHq	3802.0403200000
150	Bas -2-35	Dictionary primary preceded by one digit followed by common postfixes <([EN-3] Names-en-c.adf)>	388,287,120	8mohatued	3882.8712000000
			TOTAL TIME REQUIRED FOR WHOLE BLOCK 3 :		22 hours 42 minutes
		BLOCK#4			
151	Bas -3-01	Dictionary primary with a non-alphanumeric symbol inserted search ([EN-4] General-1-en-c.adf)	394,373,560	don(nells	3943.7356000000
152	Bas -3-01	Dictionary primary with a non-alphanumeric symbol inserted search ([EN-4] General-2-en-c.adf)	413,187,040	sali@nous	4131.8704000000
153	Bas -2-34	Dictionary primary preceded by common prefixes and followed by a one digit search <([EN-4] General-1-en-c.adf)>	456,572,400	coencodement 8	4565.7240000000
154	Bas -2-34	Dictionary primary preceded by common prefixes and followed by a one digit search <([EN-4] General-2-en-c.adf)>	456,572,400	polysiduani0	4565.7240000000
155	Bas -2-34	Dictionary primary preceded by common prefixes and followed by a one digit search <([EN-3] Names-en-c.adf)>	507,760,080	semimanlapaz 6	5077.6008000000
156	Adv -1-17	Two language-specific characters followed by four digits search	655,360,000	!*6843	6553.6000000000
157	PP 1-01	Two word concatenation without spaces search <([EN-2] Miscellaneous-en-c.adf)>	754,051,600	dynovaso	7540.5160000000
158	PP 1-02	Two word concatenation with spaces search ([EN-2] Miscellaneous-en-c.adf)	754,051,600	6root soups	7540.5160000000
159	PP 2-02	Dictionary followed by a verb or prepositional phrase followed by a Dictionary search <([EN-1] Common-en-c.adf)>	805,708,225	magicwasthel oaf	8057.0822500000

160	Adv -1-24	Dictionary primary preceded and followed by a two digit search <([EN-2] Miscellaneous-en-c.adf)>	1,098,400,000	38bens84	10984.0000000000
161	Adv -1-25	Dictionary primary followed by a four digit search ([EN-2] Miscellaneous-en-c.adf)	1,098,400,000	8hode0595	10984.0000000000
162	Adv -1-26	Dictionary primary preceded by a four digit search <([EN-2] Miscellaneous-en-c.adf)>	1,098,400,000	2619caere	10984.0000000000
163	Bas -3-04	Dictionary primary followed by a three digit search ([EN-4] General-1-en-c.adf)	1,342,860,000	hoondert637	13428.6000000000
164	Bas -3-04	Dictionary primary followed by a three digit search ([EN-4] General-2-en-c.adf)	1,342,860,000	subbued947	13428.6000000000
165	Bas -3-05	Dictionary primary preceded by a three digit search <([EN-4] General-1-en-c.adf)>	1,342,860,000	854infirmity	13428.6000000000
166	Bas -3-05	Dictionary primary preceded by a three digit search ([EN-4] General-2-en-c.adf)	1,342,860,000	387siloist	13428.6000000000
167	Bas -3-02	Dictionary primary character replacement, followed by a two digit search <([EN-3] Names-en-c.adf)>	1,376,114,400	p0nt1gg1407	13761.1440000000
168	Bas -3-03	Dictionary primary character replacement, preceded by a two digit search <([EN-3] Names-en-c.adf)>	1,376,114,400	46ov3rst0n	13761.1440000000
169	Bas -3-02	Dictionary primary character replacement, followed by a two digit search <([EN-4] General-2-en-c.adf)>	1,434,214,400	sup3rtyp351	14342.1440000000
170	Bas -3-03	Dictionary primary character replacement, preceded by a two digit search <([EN-4] General-2-en-c.adf)>	1,434,214,400	unr3cru1t4b1e79	14342.1440000000
171	Bas -3-02	Dictionary primary character replacement, followed by a two digit search <([EN-4] General-1-en-c.adf)>	1,481,848,400	impud3nt1a28	14818.4840000000
172	Bas -3-03	Dictionary primary character replacement, preceded by a two digit search <([EN-4] General-1-en-c.adf)>	1,481,848,400	15hirstb0urn3	14818.4840000000
173	Bas -3-04	Dictionary primary followed by a three digit search ([EN-3] Names-en-c.adf)	1,493,412,000	roripa954	14934.1200000000
174	Bas -3-05	Dictionary primary preceded by a three digit search ([EN-3] Names-en-c.adf)	1,493,412,000	516truyers	14934.1200000000
175	Adv -1-10	Two digits followed by three language-specific characters search	1,677,721,600	18F&}	16777.2160000000
176	Adv -1-11	Three language-specific characters followed by a two digit search	1,677,721,600	V@>29	16777.2160000000
177	Adv -1-22	Dictionary primary preceded by a two digit followed by common postfixes <([EN-4] General-1-en-c.adf)>	3,491,436,000	83gilten4u2	34914.3600000000
178	Adv -1-22	Dictionary primary preceded by a two digit followed by common postfixes <([EN-4] General-2-en-c.adf)>	3,491,436,000	69unamativeism	34914.3600000000
179	Adv -1-20	Dictionary primary followed by a two letter, language specific search ([EN-4] General-1-en-c.adf)	3,631,093,440	jowlishho	36310.9344000000
180	Adv -1-20	Dictionary primary followed by a two letter, language specific search ([EN-4] General-2-en-c.adf)	3,631,093,440	subsoilqb	36310.9344000000
181	Adv -1-21	Dictionary primary preceded by a two letter, language specific search <([EN-4] General-1-en-c.adf)>	3,631,093,440	pdconsoled	36310.9344000000

182	Adv -1-21	Dictionary primary preceded by a two letter, language specific search ([EN-4] General-2-en-c.adf)	3,631,093,440	abtintinnabulis	36310.9344000000
183	Adv -1-22	Dictionary primary preceded by a two digit followed by common postfixes <([EN-3] Names-en-c.adf)>	3,882,871,200	72pellioites	38828.7120000000
184	Bas -2-44	Social Security Number Search	4,000,000,000	078-05-1120	40000.0000000000
185	Adv -1-20	Dictionary primary followed by a two letter, language specific search ([EN-3] Names-en-c.adf)	4,038,186,048	musiciencu	40381.8604800000
186	Adv -1-21	Dictionary primary preceded by a two letter, language specific search ([EN-3] Names-en-c.adf)	4,038,186,048	hjtyronne	40381.8604800000
187	PP 2-01	Word inserted into another word search <([EN-2] Miscellaneous-en-c.adf)>	4,152,418,820	hair4hertzcut	41524.1882000000
188	Adv -1-04	All four character, language specific search	4,294,967,296	B#h{	42949.6729600000
189	Adv -1-23	Dictionary primary preceded by common prefixes and followed by a two digit search <([EN-4] General-1-en-c.adf)>	4,565,724,000	overhirudinize 52	45657.2400000000
190	Adv -1-23	Dictionary primary preceded by common prefixes and followed by a two digit search <([EN-4] General-2-en-c.adf)>	4,565,724,000	corefaced48	45657.2400000000
191	Adv -1-23	Dictionary primary preceded by common prefixes and followed by a two digit search <([EN-3] Names-en-c.adf)>	5,077,600,800	#1neilah03	50776.0080000000
192	Adv -1-24	Dictionary primary preceded and followed by a two digit search <([EN-4] General-1-en-c.adf)>	13,428,600,00 0	90hexam09	134286.0000000000
193	Adv -1-24	Dictionary primary preceded and followed by a two digit search <([EN-4] General-2-en-c.adf)>	13,428,600,00 0	65sheelfa83	134286.0000000000
194	Adv -1-25	Dictionary primary followed by a four digit search ([EN-4] General-1-en-c.adf)	13,428,600,00 0	halled8392	134286.0000000000
195	Adv -1-25	Dictionary primary followed by a four digit search ([EN-4] General-2-en-c.adf)	13,428,600,00 0	strainger2962	134286.0000000000
196	Adv -1-26	Dictionary primary preceded by a four digit search <([EN-4] General-1-en-c.adf)>	13,428,600,00 0	3952firewall	134286.0000000000
197	Adv -1-26	Dictionary primary preceded by a four digit search ([EN-4] General-2-en-c.adf)	13,428,600,00 0	2016undertake s	134286.0000000000
198	Adv -1-24	Dictionary primary preceded and followed by a two digit search <([EN-3] Names-en-c.adf)>	14,934,120,00 0	16faderman81	149341.2000000000
199	Adv -1-25	Dictionary primary followed by a four digit search ([EN-3] Names-en-c.adf)	14,934,120,00 0	manlai4614	149341.2000000000
200	Adv -1-26	Dictionary primary preceded by a four digit search ([EN-3] Names-en-c.adf)	14,934,120,00 0	1842orage	149341.2000000000
			TOTAL TIME REQUIRED FOR WHOLE BLOCK 4 :		24.8 days
		-			

APPENDIX 2: Password Files Used for Cracking

As discussed in section 4.2.2, the accounts were created in Ubuntu Linux 10.04 Long Term Support version. The /etc/shadow file was copied from Linux and the various password hashes were split up and put into individual text files. The text files were then loaded on to Distributed Network Attack for cracking. The entire list of all of the individual files loaded onto Distributed Network Attack and its contents are shown below:

main1.txt:
main1:1/uJtq9Oe$jqlVBDzjYzd4HECJ9vfQx1:14803:0:99999:7:::

main2.txt:
main2:1Uo.6TWfz$z0wqUFbEJ/FNeOFiUsSBf/:14803:0:99999:7:::

main3.txt:
main3:1vmO3xBFe$xIF/aeofcoX5obHB0jSse0:14803:0:99999:7:::

main4.txt:
main4:1OCpN0bmi$xDt/wh0rnZ4BbrihGeA9w.:14803:0:99999:7:::

main5.txt:
main5:$1$9X4I4vEk$kskQSltn6pCaPuCrHg8Si1:14803:0:99999:7:::

main6.txt:
main6:1Nd3K0/wz$Sg.rYj3DrgxL.b3f85r0W0:14803:0:99999:7:::

main7.txt:
main7:1UEoE4yZk$9WbZGIpCKMO8IqPM.Hom..:14803:0:99999:7:::

main8.txt:
main8:$1$9Ixzs6kw$hEJth50OBKJY6iAmqtCdB/:14803:0:99999:7:::

main9.txt:
main9:1E/L.kqtj$Xh5zQqBzDRjjFiOkIykOX.:14803:0:99999:7:::

main10.txt:
main10:1juUMc2Gx$A3Slct7Mbp648QR9Ad9/t0:14803:0:99999:7:::

main11.txt:
main11:1FK/E6rNz$OPi0WXCalEhoC0K0NiOP9.:14803:0:99999:7:::

main12.txt:

main12:1CSD0Tppt$0gO4c883evmn9sQRnYP6Y/:14803:0:99999:7:::

main13.txt:
main13:1p9C6AhCI$F7S10qx8r7A8gnZCGLL6V1:14803:0:99999:7:::

main14.txt:
main14:$1$583DAJNF$rN5QUZNjuex2p0VRGACkj.:14803:0:99999:7:::

main15.txt:
main15:1ZAh9sOTl$vUhDHS75AWL8r6u1ucAc.0:14803:0:99999:7:::

main16.txt:
main16:1UM10g49F$idS/4FqhLbwtvrYhkXoNR/:14803:0:99999:7:::

main17.txt:
main17:1aN46J6eT$fzg/YLa6X3bQE45/BJlaS.:14803:0:99999:7:::

main18.txt:
main18:1p4AooJ.T$CyQKj4LHyK6TN0ui1ZRd6.:14803:0:99999:7:::

main19.txt:
main19:1T5bkA4me$uEYcKeITNLgF.t9wKDoZT0:14803:0:99999:7:::

main20.txt:
main20:1Ed8Ew5TH$VaX/eiOOHOdUmH8o3DOe0/:14803:0:99999:7:::

main21.txt:
main21:1sLH1yLKF$sqL9BHOzG2v315.zAxvfl0:14803:0:99999:7:::

main22.txt:
main22:1FQb.cNHx$z3gcb69Bm8490wfgWC5NI/:14803:0:99999:7:::

main23.txt:
main23:1Rcr/wwFM$AqIn32cghgEb2sOQbRUqb1:14803:0:99999:7:::

main24.txt:
main24:1sey.DnrD$MjiTWm1OQrW6ojl1Vzje1/:14803:0:99999:7:::

main25.txt:
main25:1.kgCgb8d$bENjwurMFOyOtqCIxlqRb0:14803:0:99999:7:::

main26.txt:
main26:1RyKI2ymB$QmJizF02EuiF0NOoHbYp8.:14803:0:99999:7:::

main27.txt:
main27:1HdNI897G$Nx4fZN7vsqlzrxnwAToQe1:14803:0:99999:7:::

main28.txt:
main28:$1$5sqUJFpD$mTA0PXp9bvrNmof8wt/DH/:14803:0:99999:7:::

main29.txt:
main29:1/eD/HCSd$ROa9AirhfJrtbYwjGikEG0:14803:0:99999:7:::

main30.txt:
main30:1PSUGWdHK$jEadgIxP6qgpLuLFNbkUp.:14803:0:99999:7:::

main31.txt:
main31:1Hrsj2kD6$4Mh/yHyb2I8Zc4eMghMcG/:14803:0:99999:7:::

main32.txt:
main32:1IFnR7F7J$6p6.hJXqOuXmnhS3T.gOp0:14803:0:99999:7:::

main33.txt:
main33:1UD8Z7BAR$.8TLKviCCO9hUvk2Cn7QS/:14803:0:99999:7:::

main34.txt:
main34:1Bq8jdYWL$Fbr4VJjJtaZsiMgHg1r5.0:14803:0:99999:7:::

main35.txt:
main35:1e6FoiTze$2XeE/nlFsXEZiuv3mlmKP1:14803:0:99999:7:::

main36.txt:
main36:1MkIEqpLw$CvWbEyGPTobdshYTgVOwM/:14803:0:99999:7:::

main37.txt:
main37:$1$4l/1Li1L$.0Epn.8EngYQj28Q5wqKJ0:14803:0:99999:7:::

main38.txt:
main38:1jmw1u0Ds$lpEITDXBlKaOwGwNaoMON.:14803:0:99999:7:::

main39.txt:
main39:$1$2H3BptL2$P2j1QhTVn97ztBfWtwBZh/:14803:0:99999:7:::

main40.txt:
main40:1VEJAWFEZ$iAO/uw3/rH.9/nhNRB81V1:14803:0:99999:7:::

main41.txt:
main41:1B43.jKTn$5EPiQ8Xj1YWV/FGm4TsNL0:14803:0:99999:7:::
main42.txt:
main42:1q9D0.B/.$vz4umeODLGvaLAVUWLNrk1:14803:0:99999:7:::

main43.txt:
main43:1lwXlubXi$b7nZVGosyakD4VDYi6grA0:14803:0:99999:7:::

main44.txt:
main44:1jcb1cjHi$kWaITTAW5GdEzL75AVh.X1:14803:0:99999:7:::

main45.txt:
main45:1OIBNdCio$o5O.T/xXBqv1pYvN/rpjo1:14803:0:99999:7:::

main46.txt:
main46:1sxMJpZWz$ViAnlnnxUGLTL3vDAJAsv0:14803:0:99999:7:::

main47.txt:
main47:1tJ.VXiPw$te3jqVguGyhSYuBFyIEbT.:14803:0:99999:7:::

main48.txt:
main48:1qazUGeKV$.3P0HVV.IViVh3a4lNExP.:14803:0:99999:7:::

main49.txt:
main49:1Hql7pERD$6XrSZgCJmgQx/5zslLiSx1:14803:0:99999:7:::

main50.txt:
main50:1VNZ.bsKy$ElHe3RUGLSRF9gu/CBCxa.:14803:0:99999:7:::

main51.txt:
main51:1yr6VQjzx$QdK18RJN6zTslaj.5qM1s1:14803:0:99999:7:::

main52.txt:
main52:1/sgH26hi$GGWQAyi/eXSzthbf6/SGv/:14803:0:99999:7:::

main53.txt:
main53:1LLCf2lVB$T7EIpTAebx3o0zgS33DTJ1:14803:0:99999:7:::

main54.txt:
main54:$1$6oL5GuGD$teorCoIyRwNu8BCV/XBXm1:14803:0:99999:7:::

main55.txt:

main55:1JfLdY6IY$HPm4FCVaCAe2OMDsmWiUA/:14803:0:99999:7:::

main56.txt:
main56:1HC9rDN9y$ZgLa/nzk.qLVomBJbeh4s.:14803:0:99999:7:::
main57.txt:
main57:1iagHfCal$qrYkizQOlhLb83X9EVHI8/:14803:0:99999:7:::

main58.txt:
main58:1EZvOjYra$SviHSagLSIk56iLP01o8Y/:14803:0:99999:7:::

main59.txt:
main59:1COC27oVb$oCMj0/N.n.snbXdUJsEVF.:14803:0:99999:7:::

main60.txt:
main60:$1$8ODhEL7B$n4ynwPg75i1DvDgAE1KZt0:14803:0:99999:7:::

main61.txt:
main61:1bREDzxUy$/yW5vaBUR4HMy/O7o8Dpz.:14803:0:99999:7:::

main62.txt:
main62:1DVNOw.6Q$v7rW00aIMPrz77nG6PJn8.:14803:0:99999:7:::

main63.txt:
main63:1zaeBMSni$9QHxv1nmIVj26HTCw6Pgp.:14803:0:99999:7:::

main64.txt:
main64:1.7UWZy0/$oCeDaetD3.NwAh5CEsgyf.:14803:0:99999:7:::

main65.txt:
main65:$1$15HFuZ.E$I61aZ4OkiJp9Z0uqbFuYN1:14803:0:99999:7:::

main66.txt:
main66:1iwwlytNS$l4ALJ0pkOTtfPSD1ggQuN/:14803:0:99999:7:::

main67.txt:
main67:1/Sp46xSE$yj.lIOk06ahTQvVGm7fjt.:14803:0:99999:7:::

main68.txt:
main68:1XbzhGH2K$rHugTaPkR/jJlatfbIz6Y1:14803:0:99999:7:::

main69.txt:
main69:1cqcHpMzd$s12.tHUHUBkZe2R8GDfrR0:14803:0:99999:7:::

main70.txt:
main70:$1$7kHK0PnH$yJ5x1/KZQg2Z46R/Pan44.:14803:0:99999:7:::

main71.txt:
main71:1e1c3dzly$08yTZ6jVxR0UKQMjzaQ2h1:14803:0:99999:7:::
main72.txt:
main72:1/C8qMAQi$3/KTAQZ0JP0Hi2KfpGaq21:14803:0:99999:7:::

main73.txt:
main73:1Zr4MU962$7aQ3kL/07N5GOjRPe.2Br/:14803:0:99999:7:::

main74.txt:
main74:1oLvU/auM$wMtj/ApH/0lfq1jqkt/FK0:14803:0:99999:7:::

main75.txt:
main75:1rzoeElf8$qAjTJ81y9BjpzJd/RoTal.:14803:0:99999:7:::

main76.txt:
main76:1RoRWyKVF$fsPYY3OJfo9Ot2dsdsUd/.:14803:0:99999:7:::

main77.txt:
main77:1AK6I0BkG$ny68AN49G5.vddQCVOYG3.:14803:0:99999:7:::

main78.txt:
main78:1jGpxG9fe$yyREJPkcKFQpQqOsmy45I0:14803:0:99999:7:::

main79.txt:
main79:1JLZo21wR$.G0EhCJGLjORTuEkZM1gX0:14803:0:99999:7:::

main80.txt:
main80:1asQ1Bzb1$JDpZBWOzgzgoM7g9Qg4C80:14803:0:99999:7:::

main81.txt:
main81:1QI4iI0oI$NsbLE/GlS2.PLIp80DlRd.:14803:0:99999:7:::

main82.txt:
main82:1NGEwsnjw$GYAzXfYtTcEaG/NCGJun41:14803:0:99999:7:::

main83.txt:
main83:1tgB0.tHv$l/gvTnA2xJnikm8y7iwQs0:14803:0:99999:7:::

main84.txt:
main84:1LunE8UND$RIR3u.em69XTAUpcNFosQ/:14803:0:99999:7:::

main85.txt:
main85:$1$0pESPv1d$6cGS0J/PdM6Iy1tJKIqBz.:14803:0:99999:7:::

main86.txt:
main86:1IQq5gAED$jhSk8p9yJENnwG4NFtGIG1:14803:0:99999:7:::
main87.txt:
main87:1k5qAiLqH$GWPRTEVmM3NkWmKh5F/hk.:14803:0:99999:7:::

main88.txt:
main88:1kYQss4dp$YMTsbKdP5Z7e3SkIVe2cC/:14803:0:99999:7:::

main89.txt:
main89:1ImjyOkxh$1p.fVPDdsszsB8GoRhLfc/:14803:0:99999:7:::

main90.txt:
main90:1X15/QIgE$WysNpY41muyxuOKGwbnL6.:14803:0:99999:7:::

main91.txt:
main91:1juikhXX4$asFAEBJy4KmCQ8F5C2PEK.:14803:0:99999:7:::

main92.txt:
main92:1zhEDsS1N$fCbLN8sURHC.W.Jc9li35.:14803:0:99999:7:::

main93.txt:
main93:1i/vypP8P$74hWIU706J9zQnUPzXsjp1:14803:0:99999:7:::

main94.txt:
main94:1GleZSzY0$aORO48ah/E5Eym0mg1bkI1:14803:0:99999:7:::

main95.txt:
main95:$1$6CMYAXyz$g1uOWAC2jXXsC.EIVdPLR0:14803:0:99999:7:::

main96.txt:
main96:1cUJmDqUY$bzNTJdocoRTLelHU4A1j/.:14803:0:99999:7:::

main97.txt:
main97:1RFTVZdNt$j.sow434yX6d/8DlD4TUr1:14803:0:99999:7:::

main98.txt:
main98:1g7mKEcQj$8aQZVx8JmeZWey9AqJWog1:14803:0:99999:7:::

main99.txt:

main99:1UQiTB..u$SJUkwarkM1kzvnvt1p1kZ1:14803:0:99999:7:::

main100.txt:
main100:1uFNVMh3Q$WpacHSVFxJDf4.QSGxidm/:14803:0:99999:7:::

main101.txt:
main101:1U5kWbpnS$d/N/EC2gWqXNmQ4s0WesQ.:14803:0:99999:7:::

main102.txt:
main102:1TRqg0zPX$Qrjaa6FV6BExSno7RgQMM1:14803:0:99999:7:::

main103.txt:
main103:1BZkmvvuN$MOZOHAJ8FlXOBsL.QNP021:14803:0:99999:7:::

main104.txt:
main104:1xMe.WnnQ$mMQBJJzi6432tTsfvpPMS0:14803:0:99999:7:::

main105.txt:
main105:1U88jPkxN$mgv1wzsKvaAmpJRkCU7lb/:14803:0:99999:7:::

main106.txt:
main106:1O.xAywAA$UZmk2Kw9aFaJ0PPzNbh2B0:14803:0:99999:7:::

main107.txt:
main107:1EC1srg8t$jMDz4ZTpftEEceZ7D3Lqi/:14803:0:99999:7:::

main108.txt:
main108:$1$8ITQqR1g$JRyZdUDRML63dgSAw2XD11:14803:0:99999:7:::

main109.txt:
main109:1L5P97hdq$DlgyhAlNS7UceYAj9WBEY.:14803:0:99999:7:::

main110.txt:
main110:1X9MM2m9z$j1qUp08v2S/6KgBgbdK5T0:14803:0:99999:7:::

main111.txt:
main111:1C31tiQwg$Dr5wX8TO83.4rAtxaPn/V/:14803:0:99999:7:::

main112.txt:
main112:$1$2j3MhrkX$F0UraD1SBwq3qkqtHkGL3.:14803:0:99999:7:::

main113.txt:
main113:1jrZgrN00$6b23hy9OAJ0ZsO4uZuxja1:14803:0:99999:7:::

main114.txt:
main114:1/W5t9jxY$ccT3rDJqLkdPDP/zzgdEU/:14803:0:99999:7:::

main115.txt:
main115:1o15HIYOu$0GWXsHvju2UnMXMivOeBI1:14803:0:99999:7:::

main116.txt:
main116:$1$5EfBazQK$JhclrTLtmg/MGv/SizDy40:14803:0:99999:7:::

main117.txt:
main117:$1$3yuA2KRo$FS9BUG0Z4fh5S6TgC9Z5W/:14803:0:99999:7:::

main118.txt:
main118:1OMFF7tt1$GPLbWAHh6BsXNfmw1nXEU.:14803:0:99999:7:::

main119.txt:
main119:$1$8Ei4DtcT$bPRWoNeShW9krr5VchbRQ1:14803:0:99999:7:::

main120.txt:
main120:1GK9rgZmL$z6pKaEHTKDW0EsrvReTZh.:14803:0:99999:7:::

main121.txt:
main121:1K/3/cf6N$RoyHTCVTvyKOGyGzmMxaP0:14803:0:99999:7:::

main122.txt:
main122:$1$2P1mMrih$GaALe2v7A7/aYkoq9GD6m/:14803:0:99999:7:::

main123.txt:
main123:1P1JjD3Lf$EAUDRG/R1TjkBLmWV14rN.:14803:0:99999:7:::

main124.txt:
main124:1Wuxo9tsN$bKbK1FFwDWHomiiOYVkn/1:14803:0:99999:7:::

main125.txt:
main125:1HC1KgPVq$jRYBdyBKk5vggMMH0HiwI1:14803:0:99999:7:::

main126.txt:
main126:$1$16c.0Cqo$JtYIce5L4w6Z2D86kvJfu/:14803:0:99999:7:::

main127.txt:
main127:1A59M3YVG$JHaVPJhCZeTgU1mxZeKz91:14803:0:99999:7:::

main128.txt:
main128:1n7OEbAyB$d6K19MIm6HrLYggQShobx0:14803:0:99999:7:::

main129.txt:
main129:1YjR8qpuq$7P/OwQ2W5k3G7uwK.AoLc/:14803:0:99999:7:::

main130.txt:
main130:1.8tET9yj$pzyQFfVjPjO91J3iDsg6I.:14803:0:99999:7:::
main131.txt:
main131:1scNbvTVD$.k99N3siVOw91QcDWbV4Z/:14803:0:99999:7:::

main132.txt:
main132:1YbUsXGSr$UuENgsKQEGnX06F036Chm1:14803:0:99999:7:::

main133.txt:
main133:$1$4PURY.7C$juQ9yk/BfcofkDrOwX3Cl1:14803:0:99999:7:::

main134.txt:
main134:1fIedvzHl$d7BoTOs9o1hFoUrOgda5H.:14803:0:99999:7:::

main135.txt:
main135:1rPzOuJh2$W78zkWahGLT/PVGSs/LOS.:14803:0:99999:7:::

main136.txt:
main136:1piyb1SBQ$oZw7/3eiTI14ma8IBvkI20:14803:0:99999:7:::

main137.txt:
main137:1eTotBUgr$YtqS0O8WzL3yhAuf/Dclj1:14803:0:99999:7:::

main138.txt:
main138:1VHq95A.y$ompex9CIpAm/2VGdcLHRM/:14803:0:99999:7:::

main139.txt:
main139:1pqGYaqH0$fJ5zMfh72RtiwDDP8lFy11:14803:0:99999:7:::

main140.txt:
main140:1G7APf789$aomCwx6ydUvvR2M6Pzn30.:14803:0:99999:7:::

main141.txt:
main141:1.ZgswfSr$vDqs9ZvfeqNrdVG202gdO0:14803:0:99999:7:::

main142.txt:

main142:1hopyaZC3$stTm08OowmnmIJsKK78ln1:14803:0:99999:7:::

main143.txt:
main143:1/5xmzuup$Q.VNGQBlmR3ZWYmUg.37S1:14803:0:99999:7:::

main144.txt:
main144:$1$18hvZTyZ$OwrNAg21ZBAIOXCsR4/aU/:14803:0:99999:7:::

main145.txt:
main145:1zJB3HvQp$i97d4e68OanGZNUf1hqVY0:14803:0:99999:7:::
main146.txt:
main146:1tWcdeNY3$Rd7UYP/ImkgLCDGGCZnN7/:14803:0:99999:7:::

main147.txt:
main147:1id5BNm8q$oWBwfFT3.nWIJ/2O7Ousb/:14803:0:99999:7:::

main148.txt:
main148:1rX/6HUUs$EnwEKQ15iVmFLPrljGtuZ1:14803:0:99999:7:::

main149.txt:
main149:1I.JohYi0$dm2aDfpXvdnhibYHlvwyA0:14803:0:99999:7:::

main150.txt:
main150:1vIdsMuOF$bv4BS/BpbcQU/Ps/7JKHi.:14803:0:99999:7:::

main151.txt:
main151:1L8BK/P2J$t.K5IdcOxr4gdlo23v8aM/:14803:0:99999:7:::

main152.txt:
main152:1cSku9XJ/$mlpEeSpjqnvAVU.tUewdl.:14803:0:99999:7:::

main153.txt:
main153:1aSap9912$FA.JCHaHKi5FbXpGMBS9R1:14803:0:99999:7:::

main154.txt:
main154:1fhgbNJKQ$EGFT0uMrBlKoiATdGiygJ.:14803:0:99999:7:::

main155.txt:
main155:1sI08p35e$3PqU/7euT21rwSi9nHa8j1:14803:0:99999:7:::

main156.txt:
main156:1r/FjV7Qm$W3mDxMwxs8YSSxnKq1Y4t/:14803:0:99999:7:::

main157.txt:
main157:1VRmaYxPu$Iy08LZUpt1KGUed02b7Sn0:14803:0:99999:7:::

main158.txt:
main158:1LMgWs12U$4cJ/BqGmyGf6qH39Ss3Hm/:14803:0:99999:7:::

main159.txt:
main159:1Iq9EdkOZ$re9R31ArqYILChH1yJnUE/:14803:0:99999:7:::

main160.txt:
main160:1wfmVrgHk$LxBwFkHa1IGlRG7nrJP7f.:14803:0:99999:7:::
main161.txt:
main161:1vtxWc4SM$7uXjAVFOgCSYk.cI0RYtE/:14803:0:99999:7:::

main162.txt:
main162:1PfzpaG97$Ayiligmov52Usn7mfQ9x51:14803:0:99999:7:::

main163.txt:
main163:1Dvt4u2W8$ZezsjI5kIHxkumPgnp0Mh/:14803:0:99999:7:::

main164.txt:
main164:$1$7OY3HF8G$n9tjqf0oqk8.EqHrmxoge.:14803:0:99999:7:::

main165.txt:
main165:1K4LOHDqh$4TmWTYoGZjnFt3fa1C5.F0:14803:0:99999:7:::

main166.txt:
main166:1/g.IXmJt$y8ffMxS79nLJzoUVJHH0z/:14803:0:99999:7:::

main167.txt:
main167:1GnC8qmwI$hSwm3GRfzyJVgWmwMmzti1:14803:0:99999:7:::

main168.txt:
main168:1CZTbREts$T0oOqQBgStI6fjZC7oltv0:14803:0:99999:7:::

main169.txt:
main169:1tETe74FQ$AZCtHrqRn5AuRHlQYwb0f1:14803:0:99999:7:::

main170.txt:
main170:1x0FxYPnb$KQLdGpmHSKse.iqMroztF/:14803:0:99999:7:::

main171.txt:
main171:1PF6wu/ox$VQR/mCPQm00PmUwTCO/kd0:14803:0:99999:7:::

main172.txt:
main172:1hxoux5mn$Wx3g60osE/uK1GbRIcKxz0:14803:0:99999:7:::

main173.txt:
main173:$1$190x5OIa$CKwpr0tw7Zlbe8EKX.xaQ0:14803:0:99999:7:::

main174.txt:
main174:1t8JeORF9$5qEyjLks2peDLwgNhK4tA0:14803:0:99999:7:::

main175.txt:
main175:1c6jnY2sw$oDah6B4mcBNusL64U7TuU0:14803:0:99999:7:::
main176.txt:
main176:1Frn1LKGZ$K.Bn7wkTxpnPPLLhBw5h//:14803:0:99999:7:::

main177.txt:
main177:1DYwLGx73$PYiqjWUcVyBDOjeJZMpjS/:14803:0:99999:7:::

main178.txt:
main178:1IynJaybY$rljYgpJ1Y/atSJBCf4FVM1:14803:0:99999:7:::

main179.txt:
main179:1DQDQlP9w$iCBs9lb8RZozJy8K/oQ5q/:14803:0:99999:7:::

main180.txt:
main180:1qaFeWM/G$eSx4kU4a06.z0hdY5Os0C0:14803:0:99999:7:::

main181.txt:
main181:1tXeY7p3h$QG4e6lxGqh8N3eGuBIicU1:14803:0:99999:7:::

main182.txt:
main182:1l6jlDScs$03c2oPORMuIcaCpSek3hc/:14803:0:99999:7:::

main183.txt:
main183:$1$9GtUQhE3$HL34kNI1So3o68fDY0uJv1:14803:0:99999:7:::

main184.txt:
main184:$1$4CfsFgnc$XNj7xfDF3drcbt9juY2f3.:14803:0:99999:7:::

main185.txt:
main185:$1$1YLzpf79$Be256C79/4pFXn1Pv0uA/0:14803:0:99999:7:::

main186.txt:

main186:1HFNaeUkb$t1lZ5QcrH9iy8G/qdgTR11:14803:0:99999:7:::

main187.txt:
main187:1Bf5n1LmX$Of5sTPBVO54tiok4GkUDv0:14803:0:99999:7:::

main188.txt:
main188:1c04gsMAD$VcPsMpnLMeujNsYgcMSTW/:14803:0:99999:7:::

main189.txt:
main189:$1$5INHcQ/i$FJCp11UxTPT.O0UqTpH3g.:14803:0:99999:7:::

main190.txt:
main190:$1$0QsmVlRd$CnRAcv138kcg0AIfDg0Ij/:14803:0:99999:7:::

main191.txt:
main191:1IP/WiBO9$SdaGdfpPO1BAe5sUmIjrc0:14803:0:99999:7:::

main192.txt:
main192:1GpAjbPnI$3qy5R4oClErV4ZT3K3GSi.:14803:0:99999:7:::

main193.txt:
main193:1RL9Y9cS5$kBcOazD5gq9wpyG8WFrsw.:14803:0:99999:7:::

main194.txt:
main194:1H/Nw9If6$L0FoMpf3uiX2S.bXXRdX/0:14803:0:99999:7:::

main195.txt:
main195:1UL3CuqPs$RSgt6rtXvNXhBGDGmKToa/:14803:0:99999:7:::

main196.txt:
main196:1nMY.IqGI$KMHBb2Tr46BVCNllWW.u2/:14803:0:99999:7:::

main197.txt:
main197:1iT3.x15b$bUpiFc5Tr/xTznrUTFPQR/:14803:0:99999:7:::

main198.txt:
main198:1yfR0Y9A3$COPB3tpKTDsy053buI2wL0:14803:0:99999:7:::

main199.txt:
main199:$1$30yoDmjb$hj6FtTfFBBkWXtbd.7AtT/:14803:0:99999:7:::

main200.txt:
main200:1VoFCeqal$koutxPihpQPLMG1DGyLG40:14803:0:99999:7:::

APPENDIX 3: Password Cracking Reports Generated by Distributed Network Attack

The password cracking reports generated by Distributed Network Attack for all of the blocks are shown below.

Password Cracking Report Generated for Block 1:

DNA/PRTK Report

C:\Main Test\MainTest-passwdFiles\Block1\main9.txt
Job Status: Finished on 7/13/10 8:46:41
Commonly Registered Type: crypt user: main9
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:46:37
File Modified: 7/13/10 6:49:44
SHA 1: aa5fc73e37b0c1ebf4dc2b7654532a6c32e46fba
MD5: ae9ffdaef80dd967b3aef2749c8e2b49
Result Type:
Result: trauts
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search
C:\Main Test\MainTest-passwdFiles\Block1\main10.txt
Job Status: Finished on 7/13/10 8:46:53
Commonly Registered Type: crypt user: main10
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:46:48
File Modified: 7/13/10 6:49:58
SHA 1: d19b1b30137536ef9be7bc0698140a075e267d44
MD5: 1ed9fa57a9fb389f8cb5c63bf7bef4a8
Result Type:
Result: U%
Description: Unknown
Password Type: Password
Where Found: (ADV-1-02) All two character, language-specific search
C:\Main Test\MainTest-passwdFiles\Block1\main11.txt
Job Status: Finished on 7/13/10 8:47:07
Commonly Registered Type: crypt user: main11
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Report Date: 07/13/2010 11:27:01 Page 1
Job Started: 7/13/10 8:47:01
File Modified: 7/13/10 6:50:14

SHA 1: a168282ad70a94fbb8d5917f35050224fcac4f5f
MD5: 697d8d6915393fdfeebe8bebef52413a
Result Type:
Result: 91426
Description: Unknown
Password Type: Password
Where Found: (BAS-1-08) Five digit search
C:\Main Test\MainTest-passwdFiles\Block1\main12.txt
Job Status: Finished on 7/13/10 8:47:19
Commonly Registered Type: crypt user: main12
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:11
File Modified: 7/13/10 6:50:28
SHA 1: 79d8388daea63c8a46322e92fb6f03e3dc9ace54
MD5: 77af5d0dad71dace0e25d4bf7cf7faef
Result Type:
Result: sOcKs
Description: Unknown
Password Type: Password
Where Found: (BAS-2-19) Dictionary with two characters uppercased search
C:\Main Test\MainTest-passwdFiles\Block1\main13.txt
Job Status: Finished on 7/13/10 8:47:33
Commonly Registered Type: crypt user: main13
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:20
File Modified: 7/13/10 6:50:42
SHA 1: 82f327d6411d347d6ab4d1afb877352052437f1f
MD5: d3e1fd8802da90efa376c2d9a81ba39a
Result Type:
Result: chicle
Description: Unknown
Password Type: Password
Where Found: (BAS-2-17) Dictionary primary search
C:\Main Test\MainTest-passwdFiles\Block1\main14.txt
Job Status: Finished on 7/13/10 8:47:47
Commonly Registered Type: crypt user: main14
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:30
File Modified: 7/13/10 6:50:58
SHA 1: 3956cd8048433ee16b08d4f68a827f121257c8ce
MD5: 2b7e55992bec1cee687557c9b581b21b
Result Type:
Result: edimanoflus
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search
C:\Main Test\MainTest-passwdFiles\Block1\main15.txt

Job Status: Finished on 7/13/10 8:47:48
Commonly Registered Type: crypt user: main15
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:40
File Modified: 7/13/10 6:51:14
SHA 1: 4592a7e517dc7946125ec4c357c573bd6a64b6ad
MD5: 8c0a4a26ada8f056aa37b03bd6cef376
Result Type:
Result: aHv
Description: Unknown
Password Type: Password
Where Found: (BAS-1-06) Three letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block1\main16.txt

Job Status: Finished on 7/13/10 8:47:52
Commonly Registered Type: crypt user: main16
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:49
File Modified: 7/13/10 6:51:28
SHA 1: c674873995e003f81787d12be96b3ded6c46cb9d
MD5: 7d4f5519030dac108371aa652771e73c
No Password Found

C:\Main Test\MainTest-passwdFiles\Block1\main17.txt

Job Status: Finished on 7/13/10 8:48:02
Commonly Registered Type: crypt user: main17
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:47:59
File Modified: 7/13/10 6:51:42
SHA 1: 594bc2eec33eac48b6b6330e2f90627fb8a4ea2c
MD5: 68bef320bc19a4922631cc97a0d36a90
No Password Found

C:\Main Test\MainTest-passwdFiles\Block1\main18.txt

Job Status: Finished on 7/13/10 8:48:21
Commonly Registered Type: crypt user: main18
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:48:09
File Modified: 7/13/10 6:51:56
SHA 1: 1ed43e40e01c7b2f09c3a87b11f1a5280d924835
MD5: d50545fdc726f07eef2792b57b7c6226
Result Type:
Result: practice7
Description: Unknown
Password Type: Password
Where Found: (BAS-2-23) Dictionary primary followed by a one digit search

C:\Main Test\MainTest-passwdFiles\Block1\main1.txt
Job Status: Finished on 7/13/10 8:45:11
Commonly Registered Type: crypt user: main1
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:45:09
File Modified: 7/13/10 6:47:26
SHA 1: 45b34c55289613f9778345a1021a003c927c13ce
MD5: ab9b24fb3476d42df99315ddfaa408c1
Result Type:
Result: 9
Description: Unknown
Password Type: Password
Where Found: (BAS-1-01) One digit search
C:\Main Test\MainTest-passwdFiles\Block1\main19.txt
Job Status: Finished on 7/13/10 8:48:41
Commonly Registered Type: crypt user: main19
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:48:18
File Modified: 7/13/10 6:52:10
SHA 1: e12ecd15fd250c4d538c7b9b2e036e4b2ef6ec51
MD5: dade390a1e789a8d0dff0b56db7447fa
Result Type:
Result: 8holly
Description: Unknown
Password Type: Password
Where Found: (BAS-2-24) Dictionary primary preceded by a one digit search
C:\Main Test\MainTest-passwdFiles\Block1\main20.txt
Job Status: Finished on 7/13/10 8:48:28
Commonly Registered Type: crypt user: main20
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:48:28
File Modified: 7/13/10 6:52:30
SHA 1: 7212b49c4e00dd7c857513401a2511046682865d
MD5: 67d820245bff159113bb8f312a6fddc1
No Password Found
C:\Main Test\MainTest-passwdFiles\Block1\main21.txt
Job Status: Finished on 7/13/10 8:49:36
Commonly Registered Type: crypt user: main21
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:48:37
File Modified: 7/13/10 6:55:00
SHA 1: 3f91cfdd9382efb88007a7df8a448011cb2c4c8c
MD5: 0d22a8a3a1cb6428f60aae79cc8ded03

Result Type:
Result: joseness
Description: Unknown
Password Type: Password
Where Found: (BAS-2-21) Dictionary primary followed by common postfixes search
C:\Main Test\MainTest-passwdFiles\Block1\main22.txt
Job Status: Finished on 7/13/10 8:49:28
Commonly Registered Type: crypt user: main22
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:48:53
File Modified: 7/13/10 6:55:14
SHA 1: 410bac45cb070a5cb83114c643efc2ea7ebbc378
MD5: 4d7653166bc4448f8a3237044b2aa2b4
Result Type:
Result: 8uCE
Description: Unknown
Password Type: Password
Where Found: (BAS-2-19) Dictionary with two characters uppercased search
C:\Main Test\MainTest-passwdFiles\Block1\main23.txt
Job Status: Finished on 7/13/10 8:49:06
Commonly Registered Type: crypt user: main23
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:49:03
File Modified: 7/13/10 6:55:30
SHA 1: 12ff0eb726b645b9e8eb4c85a085cc25bc4eca65
MD5: 904df545d37336f9211165226e95e31f
No Password Found
C:\Main Test\MainTest-passwdFiles\Block1\main24.txt
Job Status: Finished on 7/13/10 8:50:29
Commonly Registered Type: crypt user: main24
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:49:13
File Modified: 7/13/10 6:55:42
SHA 1: 5e371ca864586a7d0da4961fa4e802a9b790c384
MD5: d861ef152738b9885149ee4a43fbbbb4
Result Type:
Result: outroy
Description: Unknown
Password Type: Password
Where Found: (BAS-2-22) Dictionary primary preceded by common prefixes search
C:\Main Test\MainTest-passwdFiles\Block1\main25.txt
Job Status: Finished on 7/13/10 8:50:07
Commonly Registered Type: crypt user: main25
Identified Type: *nix passwd
File Size: 60
File Version: Unknown

Job Started: 7/13/10 8:49:24
File Modified: 7/13/10 6:55:56
SHA 1: c598a7141162bafa19add77d0d272491a7fd881d
MD5: ba193157b3ad4cb5a1c5b7333bd5a843
Result Type:
Result: salmon$
Description: Unknown
Password Type: Password
Where Found: (BAS-2-27) Dictionary primary followed by a non-alphanumeric symbol search
C:\Main Test\MainTest-passwdFiles\Block1\main26.txt
Job Status: Finished on 7/13/10 8:50:50
Commonly Registered Type: crypt user: main26
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:49:34
File Modified: 7/13/10 6:56:48
SHA 1: 81f18fc6d2e03cbb47373c926426bb47031fd86c
MD5: 946a37c4426ce6302800d7096d30f6a1
Result Type:
Result: ~n2netsurfn
Description: Unknown
Password Type: Password
Where Found: (BAS-2-28) Dictionary primary preceded by a language-specific non-alphanumeric symbol
search
C:\Main Test\MainTest-passwdFiles\Block1\main27.txt
Job Status: Finished on 7/13/10 8:49:43
Commonly Registered Type: crypt user: main27
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:49:43
File Modified: 7/13/10 6:57:10
SHA 1: 17170149bdfea2f042cdac38315122c14ec62bc4
MD5: dc20f0a1a941f965739798f00add9afc
No Password Found
C:\Main Test\MainTest-passwdFiles\Block1\main28.txt
Job Status: Finished on 7/13/10 8:50:30
Commonly Registered Type: crypt user: main28
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:49:52
File Modified: 7/13/10 6:57:26
SHA 1: f3d0e9515eb78984fe83811f4149f111a8bbdbae
MD5: 6f8eba262cf03f406eae3f71320c0a37
Result Type:
Result: 517642
Description: Unknown
Password Type: Password
Where Found: (BAS-1-10) Six digit search

C:\Main Test\MainTest-passwdFiles\Block1\main2.txt
Job Status: Finished on 7/13/10 8:45:23
Commonly Registered Type: crypt user: main2
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:45:20
File Modified: 7/13/10 6:47:40
SHA 1: 0f64bb01250c25d9f1790d7d3d73a19c76ce525e
MD5: 71947e987f65b2cf66f0e13e50ee2b78
Result Type:
Result: O
Description: Unknown
Password Type: Password
Where Found: (BAS-1-02) One letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block1\main29.txt
Job Status: Finished on 7/13/10 8:50:46
Commonly Registered Type: crypt user: main29
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:02
File Modified: 7/13/10 6:57:40
SHA 1: e7ac76e51e5e3931715d072ee0f9dd5304409815
MD5: a8d9ed4b816c77114e30e568c73e686a
Result Type:
Result: 5xt2
Description: Unknown
Password Type: Password
Where Found: (BAS-2-23) Dictionary primary followed by a one digit search
C:\Main Test\MainTest-passwdFiles\Block1\main30.txt
Job Status: Finished on 7/13/10 8:50:45
Commonly Registered Type: crypt user: main30
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:12
File Modified: 7/13/10 6:57:54
SHA 1: 2bffbd1d0d5544667580b82f33e59853c7a30ebf
MD5: 0f91f8f62eef2471968c80d72fa8ca3e
Result Type:
Result: 86read
Description: Unknown
Password Type: Password
Where Found: (BAS-2-24) Dictionary primary preceded by a one digit search
C:\Main Test\MainTest-passwdFiles\Block1\main31.txt
Job Status: Finished on 7/13/10 8:50:45
Commonly Registered Type: crypt user: main31
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:21

File Modified: 7/13/10 6:58:08
SHA 1: f6c9c12656cb20da899685a390173e7a569bba53
MD5: 4cda00a6bb06ab2dbc94d20f111cc5a7
Result Type:
Result: sneakyb
Description: Unknown
Password Type: Password
Where Found: (BAS-2-25) Dictionary primary followed by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block1\main32.txt

Job Status: Finished on 7/13/10 8:51:09
Commonly Registered Type: crypt user: main32
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:31
File Modified: 7/13/10 6:58:42
SHA 1: 76181af0bda9788aef783a29a77337611a2b0948
MD5: 23b5d896cba1c0ed60b9be11fda8bd83
Result Type:
Result: xrate
Description: Unknown
Password Type: Password
Where Found: (BAS-2-26) Dictionary primary preceded by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block1\main33.txt

Job Status: Finished on 7/13/10 8:51:33
Commonly Registered Type: crypt user: main33
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:42
File Modified: 7/13/10 6:58:58
SHA 1: cdd4d086074bddb730fbf18218ac4380a4b93964
MD5: d8ca0f8c2b419a9ff3f03b0d2b8e8b6b
Result Type:
Result: foreconceive
Description: Unknown
Password Type: Password
Where Found: (BAS-2-17) Dictionary primary search

C:\Main Test\MainTest-passwdFiles\Block1\main34.txt

Job Status: Finished on 7/13/10 8:52:15
Commonly Registered Type: crypt user: main34
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:50:52
File Modified: 7/13/10 6:59:12
SHA 1: 5a092a9d24a2567a797a7c89135af7439e0e0ba3
MD5: d2cb4f92cab7fefa30f99ed5396c3ff0
Result Type:
Result: regla
Description: Unknown

Password Type: Password
Where Found: (BAS-2-17) Dictionary primary search
C:\Main Test\MainTest-passwdFiles\Block1\main35.txt
Job Status: Finished on 7/13/10 8:53:00
Commonly Registered Type: crypt user: main35
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:51:02
File Modified: 7/13/10 6:59:24
SHA 1: 9251e839629ba22cec298abde011d3ebb1acbd3a
MD5: 3e9164879c8e3c475163d500173cd770
Result Type:
Result: kcirttah
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search
C:\Main Test\MainTest-passwdFiles\Block1\main36.txt
Job Status: Finished on 7/13/10 8:52:37
Commonly Registered Type: crypt user: main36
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:51:12
File Modified: 7/13/10 6:59:40
SHA 1: 9bdbbedd25f23b69196c5f049d9cff2e7514aa44
MD5: e4dfd9f539b9a994f94e00330c385724
Result Type:
Result: esnenaoros
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search
C:\Main Test\MainTest-passwdFiles\Block1\main37.txt
Job Status: Finished on 7/13/10 8:52:43
Commonly Registered Type: crypt user: main37
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:51:22
File Modified: 7/13/10 6:59:56
SHA 1: 451080d73288acdc10df64a929bc50b7557f1a13
MD5: 2e4cd4d9d276c3813c8ab644e3517902
Result Type:
Result: maung
Description: Unknown
Password Type: Password
Where Found: (BAS-2-17) Dictionary primary search
C:\Main Test\MainTest-passwdFiles\Block1\main38.txt
Job Status: Finished on 7/13/10 8:53:13
Commonly Registered Type: crypt user: main38
Identified Type: *nix passwd
File Size: 60

File Version: Unknown
Job Started: 7/13/10 8:51:36
File Modified: 7/13/10 7:00:12
SHA 1: 7f47012e216de73805ff968f0f85fd9993428228
MD5: 2216b19abf820f43efae489c80cce82b
Result Type:
Result: asjak
Description: Unknown
Password Type: Password
Where Found: (BAS-2-18) Dictionary primary reverse search
C:\Main Test\MainTest-passwdFiles\Block1\main3.txt
Job Status: Finished on 7/13/10 8:45:35
Commonly Registered Type: crypt user: main3
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:45:32
File Modified: 7/13/10 6:47:58
SHA 1: 68ca8db8657eaedf65abbd27a2678b99a8df0317
MD5: f4e0d20e9c3c205939be6a1a253cc135
Result Type:
Result: 94
Description: Unknown
Password Type: Password
Where Found: (BAS-1-03) Two digit search
C:\Main Test\MainTest-passwdFiles\Block1\main39.txt
Job Status: Finished on 7/13/10 8:53:58
Commonly Registered Type: crypt user: main39
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:51:45
File Modified: 7/13/10 7:00:26
SHA 1: aa277126abb27b7f310017d2d72e778ac90d3c71
MD5: ca8913aa3b0f621cf80fe95a021aae6c
No Password Found
C:\Main Test\MainTest-passwdFiles\Block1\main40.txt
Job Status: Finished on 7/13/10 8:53:00
Commonly Registered Type: crypt user: main40
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:51:56
File Modified: 7/13/10 7:00:40
SHA 1: b65c08e57a02df12178393f9c5b6948e1fa95db9
MD5: cfc8a7397fd539e20e740fec500b2904
Result Type:
Result: 10ff
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block1\main41.txt

Job Status: Finished on 7/13/10 8:54:47
Commonly Registered Type: crypt user: main41
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:06
File Modified: 7/13/10 7:01:28
SHA 1: 92442b4d6f2b2ce590b0becd58f17187f5251a84
MD5: 5a550a036f57ba92d899edd7821281ef
Result Type:
Result: 9tea7
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block1\main42.txt
Job Status: Finished on 7/13/10 8:54:06
Commonly Registered Type: crypt user: main42
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:16
File Modified: 7/13/10 7:01:42
SHA 1: 7fc57224069c93ad2fd54891821aa5ae23ce5d69
MD5: 76d8c07d1b9a1f22538b9034c00f60c7
Result Type:
Result: sensor56
Description: Unknown
Password Type: Password
Where Found: (BAS-2-32) Dictionary primary followed by a two digits search
C:\Main Test\MainTest-passwdFiles\Block1\main43.txt
Job Status: Finished on 7/13/10 8:53:57
Commonly Registered Type: crypt user: main43
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:27
File Modified: 7/13/10 7:01:56
SHA 1: 78509152ce29d5c5dcf017d25d485bf97ca4b2a3
MD5: 807e396cff6a1d889b7759ee07938692
Result Type:
Result: 30pookie
Description: Unknown
Password Type: Password
Where Found: (BAS-2-33) Dictionary primary preceded by a two digits search
C:\Main Test\MainTest-passwdFiles\Block1\main44.txt
Job Status: Finished on 7/13/10 8:54:36
Commonly Registered Type: crypt user: main44
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:38
File Modified: 7/13/10 7:02:10

SHA 1: b95b12e95c70b66df01af148502e533ebc6fa4bd
MD5: dbb12634a7403090bb192842ac41d646
Result Type:
Result: X9825
Description: Unknown
Password Type: Password
Where Found: (ADV-1-07) One language-specific character followed by a four digit search

C:\Main Test\MainTest-passwdFiles\Block1\main45.txt
Job Status: Finished on 7/13/10 8:57:09
Commonly Registered Type: crypt user: main45
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:48
File Modified: 7/13/10 7:02:26
SHA 1: 927e8a6ca159018b2fc8ace767c06292835beb36
MD5: a006e912fa4fda02431cfc4f565a5bba
Result Type:
Result: 5sidenceites
Description: Unknown
Password Type: Password
Where Found: (BAS-2-21) Dictionary primary followed by common postfixes search

C:\Main Test\MainTest-passwdFiles\Block1\main46.txt
Job Status: Finished on 7/13/10 8:56:11
Commonly Registered Type: crypt user: main46
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:52:58
File Modified: 7/13/10 7:02:46
SHA 1: 1c5c0707954246f23cd2d1df472ecbb046b6db57
MD5: d21c1e88590a3a72aa03737bfb6328fc
Result Type:
Result: xYbed
Description: Unknown
Password Type: Password
Where Found: ---

C:\Main Test\MainTest-passwdFiles\Block1\main47.txt
Job Status: Finished on 7/13/10 8:56:46
Commonly Registered Type: crypt user: main47
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:53:08
File Modified: 7/13/10 7:03:00
SHA 1: 7435daa4161fdb9324588163cd62d307c73a9729
MD5: 3fe3be65e475fab41f28bf8b8b498e4e
Result Type:
Result: macasylums
Description: Unknown
Password Type: Password
Where Found: (BAS-2-22) Dictionary primary preceded by common prefixes search

C:\Main Test\MainTest-passwdFiles\Block1\main48.txt
Job Status: Finished on 7/13/10 8:55:46
Commonly Registered Type: crypt user: main48
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:53:18
File Modified: 7/13/10 7:03:12
SHA 1: 0b76c07090f105557e82e5bdb78b2ed16bc47940
MD5: 7d8bfb748dd5b44a9bda2f46fd731c44
Result Type:
Result: 4vies&
Description: Unknown
Password Type: Password
Where Found: (BAS-2-27) Dictionary primary followed by a non-alphanumeric symbol search
C:\Main Test\MainTest-passwdFiles\Block1\main4.txt
Job Status: Finished on 7/13/10 8:45:47
Commonly Registered Type: crypt user: main4
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:45:43
File Modified: 7/13/10 6:48:32
SHA 1: d14b5939c1c167d1896379d80a8486db68d16db5
MD5: 4b85e31bc3a0f5f5c8ce8815cf5a4479
Result Type:
Result: ?
Description: Unknown
Password Type: Password
Where Found: (ADV-1-01) All one-character, language-specific search
C:\Main Test\MainTest-passwdFiles\Block1\main49.txt
Job Status: Finished on 7/13/10 8:55:09
Commonly Registered Type: crypt user: main49
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:53:30
File Modified: 7/13/10 7:03:26
SHA 1: ff91f0b032609ee448e69983e2170f697ccfcd65
MD5: b32ec019e5695d6982fc7ef6bb37529f
Result Type:
Result: $5sterone
Description: Unknown
Password Type: Password
Where Found: (BAS-2-28) Dictionary primary preceded by a language-specific non-alphanumeric symbol
search
C:\Main Test\MainTest-passwdFiles\Block1\main50.txt
Job Status: Finished on 7/13/10 8:56:39
Commonly Registered Type: crypt user: main50
Identified Type: *nix passwd

File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:53:39
File Modified: 7/13/10 7:03:40
SHA 1: 86252d2f108af198808e7149b6725d214fa6bb25
MD5: e7ea97cb5c2d26ce8ccc62ee25826bec
Result Type:
Result: pyr%amid
Description: Unknown
Password Type: Password
Where Found: (BAS-3-01) Dictionary primary with a non-alphanumeric symbol inserted search

C:\Main Test\MainTest-passwdFiles\Block1\main5.txt

Job Status: Finished on 7/13/10 8:45:57
Commonly Registered Type: crypt user: main5
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:45:54
File Modified: 7/13/10 6:48:46
SHA 1: 06aa791a12963a80867bec75141595b2570a86fe
MD5: 7add21a0ca7ac8dfbc4ee7e63f48925d
Result Type:
Result: 173
Description: Unknown
Password Type: Password
Where Found: (BAS-1-05) Three digit search

C:\Main Test\MainTest-passwdFiles\Block1\main6.txt

Job Status: Finished on 7/13/10 8:46:08
Commonly Registered Type: crypt user: main6
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:46:06
File Modified: 7/13/10 6:49:02
SHA 1: 53b6afb1329c2776f13a99d8837d4e022760395b
MD5: 7a17fb7cb5ac3a5af1b951a7007c3bf6
Result Type:
Result: gS
Description: Unknown
Password Type: Password
Where Found: (BAS-1-04) Two letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block1\main7.txt

Job Status: Finished on 7/13/10 8:46:20
Commonly Registered Type: crypt user: main7
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:46:16
File Modified: 7/13/10 6:49:16
SHA 1: e353d69778058587de2636b769936fdcb058499a
MD5: ab12ed3954c59fad5dd2f261811f6752

Result Type:
Result: 3482
Description: Unknown
Password Type: Password
Where Found: (BAS-1-07) Four digit search

C:\Main Test\MainTest-passwdFiles\Block2\main85.txt
Job Status: Finished on 7/13/10 10:27:51
Commonly Registered Type: crypt user: main85
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:04:47
File Modified: 7/13/10 7:13:44
SHA 1: 28a09407369e8930bb9004f6ea3cc12f734d7169
MD5: 0a02d515e5112ffe3827a605bbed5aaf
Result Type:
Result: 832nail
Description: Unknown
Password Type: Password
Where Found: (BAS-3-05) Dictionary primary preceded by a three digit search

C:\Main Test\MainTest-passwdFiles\Block1\main8.txt
Job Status: Finished on 7/13/10 8:46:31
Commonly Registered Type: crypt user: main8
Identified Type: *nix passwd
File Size: 59
File Version: Unknown
Job Started: 7/13/10 8:46:27
File Modified: 7/13/10 6:49:28
SHA 1: f116ea2233f7cec8e0aa5f68a3ff962f5222ad1a
MD5: bfafb642cc90b6389cbe6af46bcb854a
Result Type:
Result: privs
Description: Unknown
Password Type: Password
Where Found: (BAS-2-17) Dictionary primary search

Password Cracking Report Generated for Block 2:

DNA/PRTK Report

C:\Main Test\MainTest-passwdFiles\Block2\main100.txt
Job Status: Finished on 7/13/10 10:56:13
Commonly Registered Type: crypt user: main100
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 9:10:29
File Modified: 7/13/10 7:17:32
SHA 1: eb64456e92adafd027664c1b49b48443d883afd6
MD5: bda9c827cfdae24e7a1c2e3f47d1af6c
No Password Found

C:\Main Test\MainTest-passwdFiles\Block2\main51.txt

Job Status: Finished on 7/13/10 8:59:04
Commonly Registered Type: crypt user: main51
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:58:10
File Modified: 7/13/10 7:03:58
SHA 1: b27d22ca5ec043c3a201899b8a567918f5fe67fc
MD5: d2137fcf6a8be4fbc6d685d888ce2cdc
Result Type:
Result: conpaL
Description: Unknown
Password Type: Password
Where Found: ---

C:\Main Test\MainTest-passwdFiles\Block2\main52.txt
Job Status: Finished on 7/13/10 8:59:23
Commonly Registered Type: crypt user: main52
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:58:22
File Modified: 7/13/10 7:04:16
SHA 1: ea73dd1383156e704dd0d072ea468d88e260cbd6
MD5: 1cc8bf1f930c85cc57f98e22bb20d721
No Password Found

C:\Main Test\MainTest-passwdFiles\Block2\main53.txt
Job Status: Finished on 7/13/10 9:01:31
Commonly Registered Type: crypt user: main53
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:58:32
File Modified: 7/13/10 7:04:30
SHA 1: 956eb22df67e11d71670a1c20877f17bec9c345d
MD5: 5173c0b4e4daecf4347deb74cb206aae
Result Type:
Result: 4wrightb
Description: Unknown
Password Type: Password
Where Found: (BAS-2-25) Dictionary primary followed by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block2\main54.txt
Job Status: Finished on 7/13/10 10:26:06
Commonly Registered Type: crypt user: main54
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:58:43
File Modified: 7/13/10 7:04:44
SHA 1: 1bf0715e8d0e1066eb333cb27d683b33d2e8d7fe
MD5: 18db52a59f349389eb31944b6a1c255e
Result Type:

Result: znetcom
Description: Unknown
Password Type: Password
Where Found: (BAS-2-26) Dictionary primary preceded by a one letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block2\main55.txt
Job Status: Finished on 7/13/10 9:06:53
Commonly Registered Type: crypt user: main55
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:58:53
File Modified: 7/13/10 7:05:00
SHA 1: 8e7fe5d1127da39d66a25fea566044cc48b35abe
MD5: a12f4c9b96b36b165f4acc2edd0acdf7
Result Type:
Result: 5missioners
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main56.txt
Job Status: Finished on 7/13/10 9:12:57
Commonly Registered Type: crypt user: main56
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:03
File Modified: 7/13/10 7:05:16
SHA 1: 18b48a9505466088832094868a44b146ce502d8f4
MD5: 98b5bfaa8fed385b2c78d4497ffa6e73
Result Type:
Result: tAbU
Description: Unknown
Password Type: Password
Where Found: (BAS-2-01) Four letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block2\main57.txt
Job Status: Finished on 7/13/10 9:06:04
Commonly Registered Type: crypt user: main57
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:12
File Modified: 7/13/10 7:05:28
SHA 1: b1fb601535abf8b665d1a6f1f5d27b4976d0eeeb
MD5: e82e243bec527b577214b1a3b9bc9ffc
Result Type:
Result: nonnaturfot9
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main58.txt
Job Status: Finished on 7/13/10 9:50:06
Commonly Registered Type: crypt user: main58

Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:27
File Modified: 7/13/10 7:05:40
SHA 1: 09dd5c93b9c0883370a21caa44e573285c6084be
MD5: d4164e578aedeab73d59d2a41092f0ec
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main59.txt
Job Status: Finished on 7/13/10 10:21:44
Commonly Registered Type: crypt user: main59
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:37
File Modified: 7/13/10 7:05:54
SHA 1: 7ac3b911b1905d98dd91b2e352a92ab2797ac884
MD5: e16723549e37ad90fd1b705b60430338
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main60.txt
Job Status: Finished on 7/13/10 9:06:20
Commonly Registered Type: crypt user: main60
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:47
File Modified: 7/13/10 7:06:10
SHA 1: 6a68c6dde2488775463014394e8dadb76a522eda
MD5: 30662bb77bdab2e85672dc5d011d3c3a
Result Type:
Result: 6li0n8
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main61.txt
Job Status: Finished on 7/13/10 9:07:51
Commonly Registered Type: crypt user: main61
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 8:59:56
File Modified: 7/13/10 7:07:08
SHA 1: b814f89f3ce71bb48f658ae1d3ee703778169c46
MD5: 06a9a2d0db329ba7cc9416099d3d3727
Result Type:
Result: 15o1nt
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main62.txt
Job Status: Finished on 7/13/10 9:46:34

Commonly Registered Type: crypt user: main62
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:06
File Modified: 7/13/10 7:07:22
SHA 1: 7f7b7221f11ba22c9999bbd48a80f329e69e9157
MD5: 84aed55dc3b5f42212fe358394175076
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main63.txt
Job Status: Finished on 7/13/10 9:07:31
Commonly Registered Type: crypt user: main63
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:18
File Modified: 7/13/10 7:07:40
SHA 1: 0cd0109c481e4d00b12ee5f2e1a23d5160f67585
MD5: 99c949c366988df9d13542aa57bdb6c2
Result Type:
Result: 5945259
Description: Unknown
Password Type: Password
Where Found: (BAS-2-08) Seven digit search
C:\Main Test\MainTest-passwdFiles\Block2\main64.txt
Job Status: Finished on 7/13/10 10:25:01
Commonly Registered Type: crypt user: main64
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:29
File Modified: 7/13/10 7:08:04
SHA 1: a38055101278df83d55e9359d0e608f41a65fc27
MD5: 509751d9c55b0dc3dbb96a12433affb8
Result Type:
Result: heNNigar
Description: Unknown
Password Type: Password
Where Found: (BAS-2-19) Dictionary with two characters uppercased search
C:\Main Test\MainTest-passwdFiles\Block2\main65.txt
Job Status: Finished on 7/13/10 9:02:57
Commonly Registered Type: crypt user: main65
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:39
File Modified: 7/13/10 7:08:20
SHA 1: c8a3c74590360de8f71321502e6343b27592501c
MD5: 83ec2546c1b406dbad5747e40461b6f9
Result Type:
Result: 05usite3
Description: Unknown

Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main66.txt
Job Status: Finished on 7/13/10 9:25:51
Commonly Registered Type: crypt user: main66
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:50
File Modified: 7/13/10 7:08:32
SHA 1: 525bba273aa0c575088f3149f9db25f93ea6bdb6
MD5: b9f963f000c674c423ef87fbd59e4fcb
Result Type:
Result: 6ulate27
Description: Unknown
Password Type: Password
Where Found: (BAS-2-32) Dictionary primary followed by a two digits search
C:\Main Test\MainTest-passwdFiles\Block2\main67.txt
Job Status: Finished on 7/13/10 9:31:44
Commonly Registered Type: crypt user: main67
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:00:59
File Modified: 7/13/10 7:08:46
SHA 1: 7d6198e31890544bfec908939c036e58c353ee74
MD5: d5b85be5bfe6fbf276c74ec66f92da57
Result Type:
Result: 48illegal
Description: Unknown
Password Type: Password
Where Found: (BAS-2-33) Dictionary primary preceded by a two digits search
C:\Main Test\MainTest-passwdFiles\Block2\main68.txt
Job Status: Finished on 7/13/10 10:02:28
Commonly Registered Type: crypt user: main68
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:01:10
File Modified: 7/13/10 7:09:00
SHA 1: 1530da94959bf85c313245461f78d0318dc84754
MD5: 6134726240b243ce820e299ef46b4f36
Result Type:
Result: gelosin8
Description: Unknown
Password Type: Password
Where Found: (BAS-2-23) Dictionary primary followed by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main69.txt
Job Status: Finished on 7/13/10 10:10:54
Commonly Registered Type: crypt user: main69
Identified Type: *nix passwd
File Size: 60

File Version: Unknown
Job Started: 7/13/10 9:01:19
File Modified: 7/13/10 7:09:16
SHA 1: dd4576cd71a38b6975c63482883d28a1bf702784
MD5: 70075f8f15326fdccd51a0c659c7cff8
Result Type:
Result: undercapitalised3
Description: Unknown
Password Type: Password
Where Found: (BAS-2-23) Dictionary primary followed by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main70.txt
Job Status: Finished on 7/13/10 10:03:46
Commonly Registered Type: crypt user: main70
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:01:29
File Modified: 7/13/10 7:09:28
SHA 1: d7f30829f1df391ae879fb078b7d310c592ef325
MD5: 25aaa4ce80a533d61749000f1be06132
Result Type:
Result: 7hagiographic
Description: Unknown
Password Type: Password
Where Found: (BAS-2-24) Dictionary primary preceded by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main71.txt
Job Status: Finished on 7/13/10 10:17:39
Commonly Registered Type: crypt user: main71
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:01:39
File Modified: 7/13/10 7:09:42
SHA 1: 3ecce4ada0a665a6b780ca8f658f195082a172ea
MD5: 0430b8e78bb3dfb7990992d3c9808010
Result Type:
Result: 6unoffendable
Description: Unknown
Password Type: Password
Where Found: (BAS-2-24) Dictionary primary preceded by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main72.txt
Job Status: Finished on 7/13/10 10:37:21
Commonly Registered Type: crypt user: main72
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:01:49
File Modified: 7/13/10 7:09:58
SHA 1: bd351837f485741191b23585b0b4da165ad025bb
MD5: 3159ece2f4ad40ddaa3f335a5f531652
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main73.txt

Job Status: Finished on 7/13/10 10:31:05
Commonly Registered Type: crypt user: main73
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:01:58
File Modified: 7/13/10 7:10:12
SHA 1: 8f5f6afddc3d942744ea79c18e41ee3fb57f185e
MD5: e1c798f6070bcb8a96615d743e2a1795
Result Type:
Result: th3na
Description: Unknown
Password Type: Password
Where Found: (BAS-2-20) Dictionary primary character replacements search
C:\Main Test\MainTest-passwdFiles\Block2\main74.txt
Job Status: Finished on 7/13/10 10:11:19
Commonly Registered Type: crypt user: main74
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:02:08
File Modified: 7/13/10 7:10:26
SHA 1: 645e37a26268d55567abfc019d6423d475a1080b
MD5: 835d0a9610995a13a4c8c53d61b86728
Result Type:
Result: ideoLaTry
Description: Unknown
Password Type: Password
Where Found: (BAS-2-19) Dictionary with two characters uppercased search
C:\Main Test\MainTest-passwdFiles\Block2\main75.txt
Job Status: Finished on 7/13/10 10:36:40
Commonly Registered Type: crypt user: main75
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:02:19
File Modified: 7/13/10 7:10:54
SHA 1: c8e5ef57ad2c361ebb03cc3d0b445ac8f737c0af
MD5: b6784b9ab25c654a6ae3120cfcd32eef
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main76.txt
Job Status: Finished on 7/13/10 10:31:01
Commonly Registered Type: crypt user: main76
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:02:28
File Modified: 7/13/10 7:11:08
SHA 1: 7097fe89dc3b2c93f0c01bbae30c0b713da4ae85
MD5: 22c1777edc0a62e3ffe88ac5ad3d34a6
Result Type:
Result: schleifer5

Description: Unknown
Password Type: Password
Where Found: (BAS-2-23) Dictionary primary followed by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main77.txt
Job Status: Finished on 7/13/10 9:56:10
Commonly Registered Type: crypt user: main77
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:02:40
File Modified: 7/13/10 7:11:20
SHA 1: 099b52276bf8c7a16a61b14466b9caf3a56fe517
MD5: 00ffcde42851ea6227a8cf9ef2587e65
Result Type:
Result: 4monro
Description: Unknown
Password Type: Password
Where Found: (BAS-2-24) Dictionary primary preceded by a one digit search
C:\Main Test\MainTest-passwdFiles\Block2\main78.txt
Job Status: Finished on 7/13/10 10:17:15
Commonly Registered Type: crypt user: main78
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:02:50
File Modified: 7/13/10 7:11:34
SHA 1: 6b2cb44a72b05986b03e5ca18630e53dfbdde2f7
MD5: a7f23756c569662b6614461d6ad1bc04
Result Type:
Result: slateYaRd
Description: Unknown
Password Type: Password
Where Found: (BAS-2-19) Dictionary with two characters uppercased search
C:\Main Test\MainTest-passwdFiles\Block2\main79.txt
Job Status: Finished on 7/13/10 10:38:08
Commonly Registered Type: crypt user: main79
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:02
File Modified: 7/13/10 7:11:48
SHA 1: 367ba7165adb111df77ce7e6d21faf353b69bfb3
MD5: 0e5ec68cf1efa04ab0254ecbd37f1607
Result Type:
Result: c&G
Description: Unknown
Password Type: Password
Where Found: (ADV-1-03) All three-character, language-specific search
C:\Main Test\MainTest-passwdFiles\Block2\main80.txt
Job Status: Finished on 7/13/10 10:37:16
Commonly Registered Type: crypt user: main80
Identified Type: *nix passwd

File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:12
File Modified: 7/13/10 7:12:04
SHA 1: a0eea95b2312e30d06b59c80fa6492f5f595a286
MD5: 23406f4392c3f9e6eb2155349e538e30
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main81.txt
Job Status: Finished on 7/13/10 10:12:46
Commonly Registered Type: crypt user: main81
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:23
File Modified: 7/13/10 7:12:52
SHA 1: 7d0ad0ac3b19f27b6adf5edbd90f958a90497612
MD5: edbacbf71475950603ae962e2dbf262e
Result Type:
Result: 92ars3nal
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main82.txt
Job Status: Finished on 7/13/10 10:42:44
Commonly Registered Type: crypt user: main82
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:32
File Modified: 7/13/10 7:13:04
SHA 1: e64025927ab7a6f64d04399c557ccf3b465c381f
MD5: ced6802baf313ff5ed994ebcea1a39fd
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main83.txt
Job Status: Finished on 7/13/10 10:31:07
Commonly Registered Type: crypt user: main83
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:43
File Modified: 7/13/10 7:13:16
SHA 1: a0117c6b1d370bf5e3bd785db9b20541acb68b1f
MD5: e58de4ae29d1ed5f07d4834328a08667
Result Type:
Result: eth#ics
Description: Unknown
Password Type: Password
Where Found: (BAS-3-01) Dictionary primary with a non-alphanumeric symbol inserted search
C:\Main Test\MainTest-passwdFiles\Block2\main84.txt
Job Status: Finished on 7/13/10 10:25:48

Commonly Registered Type: crypt user: main84
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:03:55
File Modified: 7/13/10 7:13:32
SHA 1: c4a3e7fa9ed69c93d1a9157bdb3ba58ce1fd13a4
MD5: dfa5d5346d8e4ccc9a609a0964d68b0e
Result Type:
Result: key395
Description: Unknown
Password Type: Password
Where Found: (BAS-3-04) Dictionary primary followed by a three digit search
C:\Main Test\MainTest-passwdFiles\Block2\main87.txt
Job Status: Finished on 7/13/10 10:43:50
Commonly Registered Type: crypt user: main87
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:07:48
File Modified: 7/13/10 7:14:10
SHA 1: f2923b56f6ebbf7e5854ea73d0569cfbe3ba03f8
MD5: 4dd41d11c963306caffcce9927286872
Result Type:
Result: mechaniccorrado
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main88.txt
Job Status: Finished on 7/13/10 10:37:47
Commonly Registered Type: crypt user: main88
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:08:01
File Modified: 7/13/10 7:14:24
SHA 1: 710dad8f98b0be79c82b90ec2b14ab1e145504a6
MD5: 35660d60b90d5526860a40579c271068
Result Type:
Result: oranges tie
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block2\main89.txt
Job Status: Finished on 7/13/10 10:49:36
Commonly Registered Type: crypt user: main89
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:08:12
File Modified: 7/13/10 7:14:40
SHA 1: 6e32d705d52c842ac10785966fe22f1293afec83

MD5: df7c66c8b6b8528176d423149260212a
Result Type:
Result: gubioa2b
Description: Unknown
Password Type: Password
Where Found: (BAS-2-21) Dictionary primary followed by common postfixes search
C:\Main Test\MainTest-passwdFiles\Block2\main90.txt
Job Status: Finished on 7/13/10 10:42:37
Commonly Registered Type: crypt user: main90
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:08:23
File Modified: 7/13/10 7:14:52
SHA 1: 57a78b3711b997f29491c2a337bce79425709572
MD5: acfc36ae724f1995a2c6f71cb7335adf
Result Type:
Result: subjing
Description: Unknown
Password Type: Password
Where Found: (BAS-2-21) Dictionary primary followed by common postfixes search
C:\Main Test\MainTest-passwdFiles\Block2\main91.txt
Job Status: Finished on 7/13/10 10:57:53
Commonly Registered Type: crypt user: main91
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:08:34
File Modified: 7/13/10 7:15:08
SHA 1: ce13adcb4822e17ca397de0d475f15da1b6f47c3
MD5: 8994bf590e161f96b55d72700ac67bb3
No Password Found
C:\Main Test\MainTest-passwdFiles\Block2\main92.txt
Job Status: Finished on 7/13/10 10:48:38
Commonly Registered Type: crypt user: main92
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:08:46
File Modified: 7/13/10 7:15:22
SHA 1: 63a740c5e9d5426ce40702ca275092035787ce8c
MD5: 11c84884ba25b2e7538e3aa839c84525
Result Type:
Result: nauenbergers
Description: Unknown
Password Type: Password
Where Found: (BAS-2-21) Dictionary primary followed by common postfixes search
C:\Main Test\MainTest-passwdFiles\Block2\main93.txt
Job Status: Finished on 7/13/10 11:02:30
Commonly Registered Type: crypt user: main93
Identified Type: *nix passwd
File Size: 60

File Version: Unknown
Job Started: 7/13/10 9:08:57
File Modified: 7/13/10 7:15:38
SHA 1: e01191b58211f68f7a43b690bc8b691aeb270db7
MD5: ddb0b13503bae4b95c4a2932b690ecb4
Result Type:
Result: posthattr
Description: Unknown
Password Type: Password
Where Found: (BAS-2-22) Dictionary primary preceded by common prefixes search

C:\Main Test\MainTest-passwdFiles\Block2\main94.txt

Job Status: Finished on 7/13/10 10:40:27
Commonly Registered Type: crypt user: main94
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:09:06
File Modified: 7/13/10 7:15:50
SHA 1: 37853133876f24312670daf6f13d2924b1779455
MD5: 8d794b45f8ae1afb316627b4bd5ec6d5
Result Type:
Result: disstomps
Description: Unknown
Password Type: Password
Where Found: (BAS-2-22) Dictionary primary preceded by common prefixes search

C:\Main Test\MainTest-passwdFiles\Block2\main95.txt

Job Status: Finished on 7/13/10 10:53:58
Commonly Registered Type: crypt user: main95
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:09:16
File Modified: 7/13/10 7:16:10
SHA 1: 9c6d2be5bb3ad2468df28fc1a013c41e9f6e12c1
MD5: 4dd0e78b0b2222765dd40e232b60e48d
Result Type:
Result: kitsipki^
Description: Unknown
Password Type: Password
Where Found: (BAS-2-27) Dictionary primary followed by a non-alphanumeric symbol search

C:\Main Test\MainTest-passwdFiles\Block2\main96.txt

Job Status: Finished on 7/13/10 10:51:14
Commonly Registered Type: crypt user: main96
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:09:29
File Modified: 7/13/10 7:16:28
SHA 1: 6441f504a7bb93d24b2fa2d097afa3e21f30ab64
MD5: 771f61b805203855649dcd15a37556a6
Result Type:

Result: trapezoidal@
Description: Unknown
Password Type: Password
Where Found: (BAS-2-27) Dictionary primary followed by a non-alphanumeric symbol
search

C:\Main Test\MainTest-passwdFiles\Block2\main97.txt
Job Status: Finished on 7/13/10 10:37:49
Commonly Registered Type: crypt user: main97
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:09:48
File Modified: 7/13/10 7:16:52
SHA 1: 835439f778abf7ba93bfd28827aa8351db2f2e21
MD5: dc1544285108b4e486b2d222e005933c
Result Type:
Result: :frontopolar
Description: Unknown
Password Type: Password
Where Found: (BAS-2-28) Dictionary primary preceded by a language-specific non-alphanumeric symbol
search

C:\Main Test\MainTest-passwdFiles\Block2\main98.txt
Job Status: Finished on 7/13/10 11:02:58
Commonly Registered Type: crypt user: main98
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:10:07
File Modified: 7/13/10 7:17:04
SHA 1: 83e33e5e5350f3ba851ab0ea5c977a330fd2dc5c
MD5: 665268e0af9562136eef94f2bcbab250
Result Type:
Result: |syndesis
Description: Unknown
Password Type: Password
Where Found: (BAS-2-28) Dictionary primary preceded by a language-specific non-alphanumeric symbol
search

C:\Main Test\MainTest-passwdFiles\Block2\main99.txt
Job Status: Finished on 7/13/10 11:00:43
Commonly Registered Type: crypt user: main99
Identified Type: *nix passwd
File Size: 60
File Version: Unknown
Job Started: 7/13/10 9:10:19
File Modified: 7/13/10 7:17:18
SHA 1: efcbd16e8fcecf9bccd6085aa3ea0eb289e28c31
MD5: 6fc52dca30958ca25776338911ee88bd
Result Type:
Result: antikillone
Description: Unknown
Password Type: Password

Where Found: (BAS-2-22) Dictionary primary preceded by common prefixes search

Password Cracking Report Generated for Block 3:

DNA/PRTK Report

C:\Main Test\MainTest-passwdFiles\Block3\main101.txt

Job Status: Finished on 7/13/10 13:21:33
Commonly Registered Type: crypt user: main101
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:31:48
File Modified: 7/13/10 7:18:12
SHA 1: 68eea9107bc429d65feb7212a36f4ecf74d36d66
MD5: 175a366ebee76f7f3b1c7050609357f4
Result Type:
Result: <lemayne
Description: Unknown
Password Type: Password
Where Found: (BAS-2-28) Dictionary primary preceded by a language-specific non-alphanumeric symbol
search

C:\Main Test\MainTest-passwdFiles\Block3\main102.txt

Job Status: Finished on 7/13/10 16:16:20
Commonly Registered Type: crypt user: main102
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:32:04
File Modified: 7/13/10 7:19:34
SHA 1: 1f32ec49f5cc891d15927cad01d329e62190b849
MD5: 02c1d97b0411b4c3edad52b59b8e555c
Result Type:
Result: 57isobared
Description: Unknown
Password Type: Password
Where Found: ---

C:\Main Test\MainTest-passwdFiles\Block3\main103.txt

Job Status: Finished on 7/13/10 16:35:42
Commonly Registered Type: crypt user: main103
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:32:15
File Modified: 7/13/10 7:19:48
SHA 1: dea0e5dbea9ed0652624b5049a7d75522a7730d5
MD5: b0e9dfc17f1b3a9b265c6311b3e77d38
Result Type:
Result: individualbc
Description: Unknown
Password Type: Password

Where Found: (ADV-1-20) Dictionary primary followed by a two letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main104.txt
Job Status: Finished on 7/13/10 18:45:03
Commonly Registered Type: crypt user: main104
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:32:27
File Modified: 7/13/10 7:20:08
SHA 1: da373fc7f7f922cc46419854428f6ae97542438d
MD5: dcb7cbcf28fb9c3eadff241835366124
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main105.txt
Job Status: Finished on 7/13/10 19:01:19
Commonly Registered Type: crypt user: main105
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:32:42
File Modified: 7/13/10 7:20:28
SHA 1: 22c07b01042c45e6be2409170d816813c0813ef5
MD5: 6a7bd996cc9ddb65d57a77ef51065d49
Result Type:
Result: lasernight
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main106.txt
Job Status: Finished on 7/13/10 18:09:00
Commonly Registered Type: crypt user: main106
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:32:53
File Modified: 7/13/10 7:20:40
SHA 1: 8dcee0906c798fb18105aff32cbf967a779c8d32
MD5: e112ec31ffe2b664051fe083a31508c3
Result Type:
Result: g#593
Description: Unknown
Password Type: Password
Where Found: (ADV-1-09) Two language-specific characters followed by a three digit search
C:\Main Test\MainTest-passwdFiles\Block3\main107.txt
Job Status: Finished on 7/13/10 15:37:26
Commonly Registered Type: crypt user: main107
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:33:05
File Modified: 7/13/10 7:20:52

SHA 1: 48355f32b875100ed4edb558dc7b68f4cabac29c
MD5: 0579bb1302144342104f41506bb7bd46
Result Type:
Result: creuxu
Description: Unknown
Password Type: Password
Where Found: (BAS-2-25) Dictionary primary followed by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block3\main108.txt
Job Status: Finished on 7/13/10 16:36:11
Commonly Registered Type: crypt user: main108
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:33:16
File Modified: 7/13/10 7:21:06
SHA 1: 860c7b38d849484b78fb7d15741a05e45e737e29
MD5: d48f56ef4de6e91181d74f980a5a11fa
Result Type:
Result: rioritya
Description: Unknown
Password Type: Password
Where Found: (BAS-2-25) Dictionary primary followed by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block3\main109.txt
Job Status: Finished on 7/13/10 20:01:49
Commonly Registered Type: crypt user: main109
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:33:30
File Modified: 7/13/10 7:21:20
SHA 1: 7f650526d82e7565e41e052713a6000654e62826
MD5: cf97cbdb5fa5707c45e97f1faa4d7055
Result Type:
Result: venprisonen
Description: Unknown
Password Type: Password
Where Found: (BAS-2-26) Dictionary primary preceded by a one letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block3\main110.txt
Job Status: Finished on 7/13/10 17:23:28
Commonly Registered Type: crypt user: main110
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:33:41
File Modified: 7/13/10 7:21:34
SHA 1: af438647c08ebb5676fe8ef8501f6e65aa934f0e
MD5: 5a47aee35cd5c7adb51d26bc8afdd293
Result Type:
Result: gseech
Description: Unknown

Password Type: Password
Where Found: (BAS-2-26) Dictionary primary preceded by a one letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main111.txt
Job Status: Finished on 7/13/10 20:15:10
Commonly Registered Type: crypt user: main111
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:33:57
File Modified: 7/13/10 7:21:48
SHA 1: ec92a17cde617e04ddefeef34fce2bc287b33510
MD5: e64ae6ff58bc5de333b40cd2991df2eb
Result Type:
Result: projewels96
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main112.txt
Job Status: Finished on 7/13/10 16:47:50
Commonly Registered Type: crypt user: main112
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:34:09
File Modified: 7/13/10 7:26:28
SHA 1: 7596ac23a951dae8a991d6c3abc2eda10bd46d74
MD5: d2460016a14302308a6fecccf38505c2
Result Type:
Result: oversonB
Description: Unknown
Password Type: Password
Where Found: (BAS-2-25) Dictionary primary followed by a one letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main113.txt
Job Status: Finished on 7/13/10 13:55:28
Commonly Registered Type: crypt user: main113
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:34:22
File Modified: 7/13/10 7:26:40
SHA 1: 56dbb6ffcb9591b65131458b0d3a07f0a2b65782
MD5: 63b8178ab6c7ad63f1f032c38d691d1c
Result Type:
Result: Jcicek
Description: Unknown
Password Type: Password
Where Found: (BAS-2-26) Dictionary primary preceded by a one letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main114.txt
Job Status: Finished on 7/13/10 20:29:53
Commonly Registered Type: crypt user: main114

Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:34:34
File Modified: 7/13/10 7:26:52
SHA 1: 2309e15abcb4e2daea1397f832492469da0ef8fb
MD5: 0b132d1eacd2cf05873cb961b0c5505c
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main115.txt
Job Status: Finished on 7/13/10 12:07:47
Commonly Registered Type: crypt user: main115
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:34:46
File Modified: 7/13/10 7:27:10
SHA 1: 613dd9fa35ae8bd57159e2c18223793aa0ca4db8
MD5: ebf71f360877773c5e97e9cb5029a7a0
Result Type:
Result: 055men4
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main116.txt
Job Status: Finished on 7/13/10 20:09:49
Commonly Registered Type: crypt user: main116
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:34:59
File Modified: 7/13/10 7:27:44
SHA 1: 3669a965468c0ac258a4691a397a849852a058a0
MD5: 728ce3df019f3bdcff6c64ea4f45d327
Result Type:
Result: 67298157
Description: Unknown
Password Type: Password
Where Found: (BAS-2-13) Eight digit search
C:\Main Test\MainTest-passwdFiles\Block3\main117.txt
Job Status: Finished on 7/13/10 19:20:38
Commonly Registered Type: crypt user: main117
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:35:10
File Modified: 7/13/10 7:27:58
SHA 1: 9b5372deb933c92eca2ecee2ed2f4adf34b93185
MD5: edd7692b817401ba3a8a89b336a130be
Result Type:
Result: 5rvants167
Description: Unknown
Password Type: Password

Where Found: (BAS-3-04) Dictionary primary followed by a three digit search
C:\Main Test\MainTest-passwdFiles\Block3\main118.txt
Job Status: Finished on 7/13/10 19:45:23
Commonly Registered Type: crypt user: main118
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:35:21
File Modified: 7/13/10 7:28:12
SHA 1: 856e44ba0e4d20e172768aebf5ed79d130815130
MD5: 97f27d80fe5f16232eb15f380b37e940
Result Type:
Result: 6277polis
Description: Unknown
Password Type: Password
Where Found: (BAS-3-05) Dictionary primary preceded by a three digit search
C:\Main Test\MainTest-passwdFiles\Block3\main119.txt
Job Status: Finished on 7/13/10 21:03:21
Commonly Registered Type: crypt user: main119
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:35:31
File Modified: 7/13/10 7:28:26
SHA 1: bd24344f35f4bd072e7c454407ea69d7e18beb88
MD5: 42e8993807c0633d274ecc861fb069fa
Result Type:
Result: 7galvanometers9
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main120.txt
Job Status: Finished on 7/13/10 15:43:33
Commonly Registered Type: crypt user: main120
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:35:45
File Modified: 7/13/10 7:28:42
SHA 1: ded2b3bee7a6463ee7ab9967329ac36f0f79a0fa
MD5: 04d91a2e4258476caec7f48ed10c6dac
Result Type:
Result: 2surle4
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main121.txt
Job Status: Finished on 7/13/10 19:41:03
Commonly Registered Type: crypt user: main121
Identified Type: *nix passwd
File Size: 61
File Version: Unknown

Job Started: 7/13/10 11:35:55
File Modified: 7/13/10 7:29:40
SHA 1: 25c44280fc8117920a3e468a167da21472d48b3f
MD5: 467be1d33bc8c289e2c126f0530f7341
Result Type:
Result: floorings73
Description: Unknown
Password Type: Password
Where Found: (BAS-2-32) Dictionary primary followed by a two digits search
C:\Main Test\MainTest-passwdFiles\Block3\main122.txt
Job Status: Finished on 7/13/10 17:41:01
Commonly Registered Type: crypt user: main122
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:36:07
File Modified: 7/13/10 7:29:52
SHA 1: f66f5771e92f3f58f9380a5c86ee85e6953224cd
MD5: d30445ca93d7393c5f3ce9bfd0deeed9
Result Type:
Result: ricochet24
Description: Unknown
Password Type: Password
Where Found: (BAS-2-32) Dictionary primary followed by a two digits search
C:\Main Test\MainTest-passwdFiles\Block3\main123.txt
Job Status: Finished on 7/13/10 22:24:25
Commonly Registered Type: crypt user: main123
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:36:19
File Modified: 7/13/10 7:30:12
SHA 1: e2b1405a3a0f48384e964537ccda7a6562d9d13b
MD5: f282936bc985ae895a4b157af39d2163
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main124.txt
Job Status: Finished on 7/13/10 19:46:13
Commonly Registered Type: crypt user: main124
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:36:31
File Modified: 7/13/10 7:30:24
SHA 1: 31ec3e89e1be23e67a72981a5205e378410f2fce
MD5: 669ca92ad8083015795f4b850533672a
Result Type:
Result: 64pseud
Description: Unknown
Password Type: Password
Where Found: (BAS-2-33) Dictionary primary preceded by a two digits search
C:\Main Test\MainTest-passwdFiles\Block3\main125.txt
Job Status: Finished on 7/13/10 18:41:47

Commonly Registered Type: crypt user: main125
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:36:42
File Modified: 7/13/10 7:30:36
SHA 1: 6a3a146ada738060e1f713d0d12201c6ba1a8054
MD5: b7a087bb8a9fda2791926538900140e5
Result Type:
Result: m4ll4m5
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main126.txt
Job Status: Finished on 7/13/10 22:03:42
Commonly Registered Type: crypt user: main126
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:36:54
File Modified: 7/13/10 7:30:52
SHA 1: 00373232b6aec07a4da0f1db6bb8e8e63f85243f
MD5: 1e18099e9c043fa233231ffff12a1b49
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main127.txt
Job Status: Finished on 7/13/10 22:13:36
Commonly Registered Type: crypt user: main127
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:37:06
File Modified: 7/13/10 7:31:04
SHA 1: 04ccab4936654222eb789a948fc0fff6cc64f44d
MD5: f7c52e65ec58e20cd6cc07cf51c3a615
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main128.txt
Job Status: Finished on 7/13/10 22:14:53
Commonly Registered Type: crypt user: main128
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:37:19
File Modified: 7/13/10 7:31:16
SHA 1: 97f4ae51ca4afee7d53a3665b3647474cccca9a3
MD5: 467c1d0af0f93738429d490ed5349202
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main129.txt
Job Status: Finished on 7/13/10 22:28:31
Commonly Registered Type: crypt user: main129
Identified Type: *nix passwd
File Size: 61

File Version: Unknown
Job Started: 7/13/10 11:37:30
File Modified: 7/13/10 7:31:28
SHA 1: 80bc13ad37407a3541137627e5f57c12652afb2a
MD5: 17299a33412ad768a27451073a7a9616
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main130.txt
Job Status: Finished on 7/13/10 22:04:45
Commonly Registered Type: crypt user: main130
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:37:41
File Modified: 7/13/10 7:31:42
SHA 1: 9fd2214c4ba0ce319e717111f17838e8b72a3f6a
MD5: c98c8b71519e3ef82c88ea11b8cab232
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main131.txt
Job Status: Finished on 7/13/10 18:42:20
Commonly Registered Type: crypt user: main131
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:37:50
File Modified: 7/13/10 7:31:54
SHA 1: 5f2d4e9d22082a90c33de1aa8bed4168cf8a5849
MD5: 6ec8e56cb4fffa8b961ee655ec10c775
Result Type:
Result: 5ogawara3
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main132.txt
Job Status: Finished on 7/13/10 19:58:03
Commonly Registered Type: crypt user: main132
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:38:00
File Modified: 7/13/10 7:32:08
SHA 1: 787099465f9deba3682897ac8273b0961b78e0c9
MD5: 68d4861054423a2ea44d5fa2a93a8ea8
Result Type:
Result: nkwazo92
Description: Unknown
Password Type: Password
Where Found: (BAS-2-32) Dictionary primary followed by a two digits search
C:\Main Test\MainTest-passwdFiles\Block3\main133.txt
Job Status: Finished on 7/13/10 14:51:45
Commonly Registered Type: crypt user: main133
Identified Type: *nix passwd

File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:38:10
File Modified: 7/13/10 7:32:26
SHA 1: 28f6eb209e77d5ffdc2e979f71d1df225d8b4430
MD5: 6cd6db4d5b28aad3c1c3482717823033
Result Type:
Result: 20niegel
Description: Unknown
Password Type: Password
Where Found: (BAS-2-33) Dictionary primary preceded by a two digits search
C:\Main Test\MainTest-passwdFiles\Block3\main134.txt
Job Status: Finished on 7/13/10 18:51:07
Commonly Registered Type: crypt user: main134
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:38:24
File Modified: 7/13/10 7:32:54
SHA 1: 6802e7f81d092e692fa457d4b55fe1031422933b
MD5: a6c65169feba5482af6820f83402bf2f
Result Type:
Result: 4i(D
Description: Unknown
Password Type: Password
Where Found: (ADV-1-05) One digit followed by three language-specific characters search
C:\Main Test\MainTest-passwdFiles\Block3\main135.txt
Job Status: Finished on 7/13/10 22:24:32
Commonly Registered Type: crypt user: main135
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:38:34
File Modified: 7/13/10 7:33:22
SHA 1: ba73d011b721fb01a9a7bd0c7b991935355b8604
MD5: 94817eaffe8118be8cbc842975de95a8
Result Type:
Result: {Y^8
Description: Unknown
Password Type: Password
Where Found: (ADV-1-06) Three language-specific characters followed by one digit search
C:\Main Test\MainTest-passwdFiles\Block3\main136.txt
Job Status: Finished on 7/13/10 19:15:55
Commonly Registered Type: crypt user: main136
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:38:48
File Modified: 7/13/10 7:33:42
SHA 1: a544c75380d4cbd247270de666d7ed08830f68a9

MD5: 2d9d901e6064248ce2c742d6de382f59
Result Type:
Result: lyryannn
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main137.txt
Job Status: Finished on 7/13/10 22:50:06
Commonly Registered Type: crypt user: main137
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:39:00
File Modified: 7/13/10 7:34:04
SHA 1: 7c808364f732ed48648872cb915ccd80f44d6310
MD5: 686331d29af43e07ffef8b8f4c9e5915
Result Type:
Result: udcj!@#
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main138.txt
Job Status: Finished on 7/13/10 22:51:47
Commonly Registered Type: crypt user: main138
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:39:13
File Modified: 7/13/10 7:34:20
SHA 1: 244fb42c30d17e898029a038f7cf69e2328b7444
MD5: ef63eba62ab10d0618d1479218376541
Result Type:
Result: 89quick24
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main139.txt
Job Status: Finished on 7/13/10 22:51:51
Commonly Registered Type: crypt user: main139
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:39:27
File Modified: 7/13/10 7:34:36
SHA 1: 9de56e234122d00e7a592eb1b081602353bd1073
MD5: 20e47d344b8e687c1fa893a89eb68cb5
Result Type:
Result: tryscer2374
Description: Unknown
Password Type: Password
Where Found: (ADV-1-25) Dictionary primary followed by a four digit search
C:\Main Test\MainTest-passwdFiles\Block3\main140.txt
Job Status: Finished on 7/13/10 15:58:10

Commonly Registered Type: crypt user: main140
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:39:37
File Modified: 7/13/10 7:34:48
SHA 1: 3efd08cf493317e9701ab9676227924667838b39
MD5: fd89520b1604fd27aad34a3f2f831b02
Result Type:
Result: 1943percolate
Description: Unknown
Password Type: Password
Where Found: (ADV-1-26) Dictionary primary preceded by a four digit search
C:\Main Test\MainTest-passwdFiles\Block3\main142.txt
Job Status: Finished on 7/13/10 19:43:53
Commonly Registered Type: crypt user: main142
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:40:10
File Modified: 7/13/10 7:36:02
SHA 1: 7c2bb9e4ffab250123e585707016fa834a7b4453
MD5: 5e4a4b534d90e5f7d75ce5a395eaf996
Result Type:
Result: 34deadensing
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main143.txt
Job Status: Finished on 7/13/10 21:16:37
Commonly Registered Type: crypt user: main143
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:40:21
File Modified: 7/13/10 7:36:14
SHA 1: d39bdae0d4d7757852438573c149bf3964e44c5d
MD5: 6333951244bb0fdab5d2a78486b62427
Result Type:
Result: 5etudeje
Description: Unknown
Password Type: Password
Where Found: (ADV-1-20) Dictionary primary followed by a two letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main144.txt
Job Status: Finished on 7/13/10 21:45:48
Commonly Registered Type: crypt user: main144
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:40:34
File Modified: 7/13/10 7:36:28

SHA 1: 3aecfca2d24d5dde5b9be22d167e3c94d133f63f
MD5: cf4accf131f237d720a0564023b1b48c
Result Type:
Result: asfaculty
Description: Unknown
Password Type: Password
Where Found: (ADV-1-21) Dictionary primary preceded by a two letter, language specific search

C:\Main Test\MainTest-passwdFiles\Block3\main145.txt
Job Status: Finished on 7/13/10 22:53:04
Commonly Registered Type: crypt user: main145
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:40:49
File Modified: 7/13/10 7:36:40
SHA 1: 293cda390724d629fafce000daaf1da983ab4880
MD5: a4276fc7c8ecba8d1c8b39989b79286c
Result Type:
Result: oshi$ka
Description: Unknown
Password Type: Password
Where Found: (BAS-3-01) Dictionary primary with a non-alphanumeric symbol inserted search

C:\Main Test\MainTest-passwdFiles\Block3\main146.txt
Job Status: Finished on 7/13/10 19:17:31
Commonly Registered Type: crypt user: main146
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:41:00
File Modified: 7/13/10 7:36:52
SHA 1: 4752235fd511c58500ab96a8ed2fa43cdaf99c4f
MD5: 2145b35256b1e870afdfacf4e8313fa9
Result Type:
Result: 2exsteties
Description: Unknown
Password Type: Password
Where Found: ---

C:\Main Test\MainTest-passwdFiles\Block3\main147.txt
Job Status: Finished on 7/13/10 22:55:14
Commonly Registered Type: crypt user: main147
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:41:10
File Modified: 7/13/10 7:37:04
SHA 1: c6f9559a86561c4a391a1f0e546a51f812eccdc3
MD5: fdc4cdce0790ca03da627711861016e1
Result Type:
Result: 5snapbackers
Description: Unknown
Password Type: Password

Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main148.txt
Job Status: Finished on 7/13/10 23:20:56
Commonly Registered Type: crypt user: main148
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:41:22
File Modified: 7/13/10 7:37:20
SHA 1: 2f845e0a4d16a3027e0965d9f339d3375080564f
MD5: 29084cf579788496d1725d0a8a1666a5
No Password Found
C:\Main Test\MainTest-passwdFiles\Block3\main149.txt
Job Status: Finished on 7/13/10 22:02:50
Commonly Registered Type: crypt user: main149
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:41:33
File Modified: 7/13/10 7:37:34
SHA 1: 2eaf59cd6465ab454dbc64ea1e03657cacf48c3d
MD5: 0a11c4c2c4fad23b32cb9bbce56b19ee
Result Type:
Result: bDtHq
Description: Unknown
Password Type: Password
Where Found: (BAS-2-02) Five letter, language specific search
C:\Main Test\MainTest-passwdFiles\Block3\main150.txt
Job Status: Finished on 7/13/10 23:36:09
Commonly Registered Type: crypt user: main150
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:41:45
File Modified: 7/13/10 7:37:46
SHA 1: 862c2df8f2d80f817297196b3d3deb3cfe1df337
MD5: 79828b50f5e68267623b2a508f65a39a
Result Type:
Result: 8mohatued
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block3\main141.txt
Job Status: Finished on 7/13/10 22:13:13
Commonly Registered Type: crypt user: main141
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/13/10 11:44:19
File Modified: 7/13/10 7:35:48
SHA 1: 33abceafd1a1d324f1af973b84455bb6d00eb593

MD5: 2bde53fe4272f8ff6a2a05e44b160d3c
Result Type:
Result: non-fyhy
Description: Unknown
Password Type: Password
Where Found: ---

Password Cracking Report Generated for Block 4:

DNA/PRTK Report

C:\Main Test\MainTest-passwdFiles\Block4\main151.txt

Job Status: Finished on 7/15/10 3:25:21
Commonly Registered Type: crypt user: main151
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:58:56
File Modified: 7/13/10 7:38:34
SHA 1: 6cda9f4e62288d2ccc46c4a817daaf1153dbeced
MD5: f12c2596d1cdc9e3f7833e6397c513c8
Result Type:
Result: don(nells
Description: Unknown
Password Type: Password
Where Found: (BAS-3-01) Dictionary primary with a non-alphanumeric symbol inserted search

C:\Main Test\MainTest-passwdFiles\Block4\main152.txt

Job Status: Finished on 7/15/10 23:03:12
Commonly Registered Type: crypt user: main152
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:59:14
File Modified: 7/13/10 7:38:50
SHA 1: c1a160f42210e0fcc8ba09e0c5057d7d3e814595
MD5: bcb9d4ad169313988d576e14be630091
Result Type:
Result: sali@nous
Description: Unknown
Password Type: Password
Where Found: (BAS-3-01) Dictionary primary with a non-alphanumeric symbol inserted search

C:\Main Test\MainTest-passwdFiles\Block4\main153.txt

Job Status: Finished on 7/15/10 2:07:12
Commonly Registered Type: crypt user: main153
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:59:25
File Modified: 7/13/10 7:39:02
SHA 1: 1a7d5fe4da42c844a6aeab41d5a9a05e9dc2cb31

MD5: 17632fca93fa27bb8d0a89c80a2ba92e
Result Type:
Result: coencodement8
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main154.txt
Job Status: Finished on 7/17/10 8:02:59
Commonly Registered Type: crypt user: main154
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:59:36
File Modified: 7/13/10 7:39:16
SHA 1: 7023ed9eb0f14174166aa4b723b8eb5341cbc459
MD5: c798bd27d3f0ba5774830f6c666f108d
Result Type:
Result: polysiduani0
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main155.txt
Job Status: Finished on 7/17/10 15:41:19
Commonly Registered Type: crypt user: main155
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:59:47
File Modified: 7/13/10 7:39:30
SHA 1: 54c1931b6fc2307ba83f9ba524861e6e8d6f792b
MD5: c1728fcecbf224fa4043f22bbd96229e
Result Type:
Result: semimanlapaz6
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main156.txt
Job Status: Finished on 7/17/10 7:14:26
Commonly Registered Type: crypt user: main156
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 8:59:59
File Modified: 7/13/10 7:39:42
SHA 1: 17fc4c0244dcefaf15ad9ded2bdd2d14357befa1
MD5: 62f18d6879676ec91942080f383abdc5
Result Type:
Result: !*6843
Description: Unknown
Password Type: Password
Where Found: (ADV-1-17) Two language-specific characters followed by four digits search
C:\Main Test\MainTest-passwdFiles\Block4\main157.txt

Job Status: Finished on 7/18/10 7:49:16
Commonly Registered Type: crypt user: main157
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:00:12
File Modified: 7/13/10 7:39:56
SHA 1: b0cbefa18958880b2f543b63f8a202657a4ad87b
MD5: 3b533e24bd2057b1e6b2ee4745be0d90
Result Type:
Result: dynovaso
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main158.txt
Job Status: Finished on 7/17/10 12:27:54
Commonly Registered Type: crypt user: main158
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:00:23
File Modified: 7/13/10 7:40:10
SHA 1: 11a0aed844214edb4b1cc0aab1dd2f8ffa1286a1
MD5: f0560feeec815fef74031d6835ae4abc
Result Type:
Result: 6root soups
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main159.txt
Job Status: Finished on 7/19/10 10:04:05
Commonly Registered Type: crypt user: main159
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:00:35
File Modified: 7/13/10 7:40:30
SHA 1: 744f5123345c57bd8efa8ca5886c6ca8c639b0ba
MD5: af0c695ce0c8fe9ab83673779142cba3
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main160.txt
Job Status: Finished on 7/16/10 17:23:46
Commonly Registered Type: crypt user: main160
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:00:45
File Modified: 7/13/10 7:40:42
SHA 1: 5f305dac26267de58489e6f6bcd02ea97b34a03f
MD5: b10b390a02c0f68f4506c3af60e01934
Result Type:
Result: 38bens84

Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main161.txt
Job Status: Finished on 7/18/10 5:06:09
Commonly Registered Type: crypt user: main161
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:00:56
File Modified: 7/13/10 7:42:24
SHA 1: 210ff0dba7610202198cc560f894bb954d36d9f2
MD5: 7917e043d6670df9a2a55eb751d2f022
Result Type:
Result: 8hode0595
Description: Unknown
Password Type: Password
Where Found: (ADV-1-25) Dictionary primary followed by a four digit search
C:\Main Test\MainTest-passwdFiles\Block4\main162.txt
Job Status: Finished on 7/16/10 2:06:11
Commonly Registered Type: crypt user: main162
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:01:06
File Modified: 7/13/10 7:43:04
SHA 1: 5b1fe380184f6c96d72c0bcf5dd39652f858a39c
MD5: 721dd4f5d1cc88ce6c895d380399a52e
Result Type:
Result: 2916caere
Description: Unknown
Password Type: Password
Where Found: (ADV-1-26) Dictionary primary preceded by a four digit search
C:\Main Test\MainTest-passwdFiles\Block4\main163.txt
Job Status: In process.
Commonly Registered Type: crypt user: main163
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:01:18
File Modified: 7/13/10 7:43:18
SHA 1: 51fb04bd265fcf82a54ce4e8af5895a076b04050
MD5: 8be7635defbae0a5c1d81777bc55cdef
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main164.txt
Job Status: Finished on 7/18/10 11:02:14
Commonly Registered Type: crypt user: main164
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:01:28
File Modified: 7/13/10 7:43:34

SHA 1: fdb6b572ed89d586d707cf7b022de9dbff7c7e3e
MD5: fbad335e8eddb60895ee7952296e2c93
Result Type:
Result: subbued947
Description: Unknown
Password Type: Password
Where Found: (BAS-3-04) Dictionary primary followed by a three digit search
C:\Main Test\MainTest-passwdFiles\Block4\main165.txt
Job Status: In process.
Commonly Registered Type: crypt user: main165
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:01:39
File Modified: 7/13/10 7:43:50
SHA 1: f27ad6e410d1e4f396d88cbab987d93bfe29563f
MD5: 6d6629fd350bf548aacf535284027cb2
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main166.txt
Job Status: Finished on 7/17/10 3:57:51
Commonly Registered Type: crypt user: main166
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:01:49
File Modified: 7/13/10 7:44:02
SHA 1: fa0194ecb75eb081f054559ff045747497b78775
MD5: 402297d9ffc46816bb513eebb2b68eed
Result Type:
Result: 387siloist
Description: Unknown
Password Type: Password
Where Found: (BAS-3-05) Dictionary primary preceded by a three digit search
C:\Main Test\MainTest-passwdFiles\Block4\main167.txt
Job Status: In process.
Commonly Registered Type: crypt user: main167
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:00
File Modified: 7/13/10 7:44:16
SHA 1: 8ed3d99332b81665b3ab0bed311e34b6b9eb126f
MD5: 08cc33328b114be5005ad415a9919ad0
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main168.txt
Job Status: In process.
Commonly Registered Type: crypt user: main168
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:11

File Modified: 7/13/10 7:44:34
SHA 1: 24eb9ce094b9b4dc25336339bce641d6e0778a18
MD5: 4547dedbfd8234badd1c6cc20d910dff
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main169.txt
Job Status: Finished on 7/19/10 19:42:00
Commonly Registered Type: crypt user: main169
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:21
File Modified: 7/13/10 7:44:48
SHA 1: ccf5963681dc00b79f4f29d72baa0ba56612441f
MD5: 59c0bfd46fbeb8738c8210d9229dabee
Result Type:
Result: sup3rtyp351
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main170.txt
Job Status: Finished on 7/20/10 5:48:47
Commonly Registered Type: crypt user: main170
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:31
File Modified: 7/13/10 7:45:00
SHA 1: d7a3137874461f3f4f17534a3bafa4003e3d3cbf
MD5: a253f6362249d109137a57061a134a73
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main171.txt
Job Status: In process.
Commonly Registered Type: crypt user: main171
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:42
File Modified: 7/13/10 7:45:14
SHA 1: 517b92606857c37391d97a22d5f813c57ec7de5b
MD5: 4f992fe4b3a6cfc9957ba274cc7a1c69
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main172.txt
Job Status: Finished on 7/20/10 5:30:36
Commonly Registered Type: crypt user: main172
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:02:53
File Modified: 7/13/10 7:45:34
SHA 1: 7b0db161c101a0dc6070ce89efb402d31d44b0e9
MD5: 7392ebec7c60bf7c2a7c417996322f0b

No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main173.txt
Job Status: Finished on 7/19/10 19:34:03
Commonly Registered Type: crypt user: main173
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:03:03
File Modified: 7/13/10 7:45:50
SHA 1: 908f877e748c282f0dd0454d62b162b39e73fca1
MD5: 0d81b582d611971b98316040c49f7f36
Result Type:
Result: roripa954
Description: Unknown
Password Type: Password
Where Found: (BAS-3-04) Dictionary primary followed by a three digit search
C:\Main Test\MainTest-passwdFiles\Block4\main174.txt
Job Status: Finished on 7/17/10 15:39:07
Commonly Registered Type: crypt user: main174
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:03:14
File Modified: 7/13/10 7:46:04
SHA 1: 76ea6be335ddd570cc1a764461ea6ea16e2bcca7
MD5: 32a465b90d0bcd8c3bc355d544c6cfe3
Result Type:
Result: 516truyers
Description: Unknown
Password Type: Password
Where Found: (BAS-3-05) Dictionary primary preceded by a three digit search
C:\Main Test\MainTest-passwdFiles\Block4\main175.txt
Job Status: Finished on 7/15/10 16:34:25
Commonly Registered Type: crypt user: main175
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:03:25
File Modified: 7/13/10 7:46:34
SHA 1: 198e108a5127840b5fdd07de26c56487e76710db
MD5: c6d4784b144f9b1cc99dadccba10e20e
Result Type:
Result: 18F&}
Description: Unknown
Password Type: Password
Where Found: (ADV-1-10) Two digits followed by three language-specific characters search
C:\Main Test\MainTest-passwdFiles\Block4\main176.txt
Job Status: Finished on 7/19/10 17:58:48
Commonly Registered Type: crypt user: main176
Identified Type: *nix passwd
File Size: 61

File Version: Unknown
Job Started: 7/14/10 9:03:36
File Modified: 7/13/10 7:46:48
SHA 1: 5894f29484b804b493a01a07c60bdcef10949057
MD5: 04db55937472addd99d069187d7b4908
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main177.txt
Job Status: In process.
Commonly Registered Type: crypt user: main177
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:03:46
File Modified: 7/13/10 7:47:02
SHA 1: 86b4914b2a8b0ac7240d3a4a11b4c9790ddc94b0
MD5: 37a79733750ae213a83441fd42ef7d88
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main178.txt
Job Status: In process.
Commonly Registered Type: crypt user: main178
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:03:57
File Modified: 7/13/10 7:47:26
SHA 1: 784a83b53aa0c0acffa7f4406e1cdb1c918d692a
MD5: d7f4788e9484ed30db4e46071e2a45a8
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main179.txt
Job Status: In process.
Commonly Registered Type: crypt user: main179
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:04:07
File Modified: 7/13/10 7:47:40
SHA 1: afff2e15b635b3f52f110b5032f80b60b034248c
MD5: 2de532b4083fad6b7415b35bf4c6e8dc
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main180.txt
Job Status: In process.
Commonly Registered Type: crypt user: main180
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:04:19
File Modified: 7/13/10 7:47:54
SHA 1: d8cf18ce7630ac27eff833c7b9b9f1ab86e998bf
MD5: 1e4e6815f77aabb1eda53450c304d470
No Password Found

C:\Main Test\MainTest-passwdFiles\Block4\main181.txt
Job Status: In process.
Commonly Registered Type: crypt user: main181
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:04:30
File Modified: 7/13/10 7:48:42
SHA 1: c30b5193d6ef75f92a2e90a5640d1cf53ddb1d1e
MD5: 38ae554218c616d19d96d7bfb9975882
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main182.txt
Job Status: In process.
Commonly Registered Type: crypt user: main182
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:04:41
File Modified: 7/13/10 7:48:56
SHA 1: 3ced0af70d367364b7603c05be98336ceda4ddd1
MD5: 012b0a6023ec9df1c2bb908aea40b251
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main183.txt
Job Status: In process.
Commonly Registered Type: crypt user: main183
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:04:53
File Modified: 7/13/10 7:49:16
SHA 1: d733ce8794fe0aebaa563ae8921b777344f038fb
MD5: 175b3e83dff289b198a48527e909acd8
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main184.txt
Job Status: Finished on 7/15/10 15:10:20
Commonly Registered Type: crypt user: main184
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:05:09
File Modified: 7/13/10 7:50:20
SHA 1: 319be674575449fcec3f1c3ef169a3dc3962881b
MD5: 9711a9265b54232dd2bcb58bad3d686e
Result Type:
Result: 078-05-1120
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main185.txt
Job Status: In process.

Commonly Registered Type: crypt user: main185
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:05:19
File Modified: 7/13/10 7:50:38
SHA 1: 1cce54c5f4db35cfbdf59080cd925a8ac774a9f1
MD5: 9d2d69cc98515e4e4024811b86e8e8fa
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main186.txt
Job Status: In process.
Commonly Registered Type: crypt user: main186
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:05:30
File Modified: 7/13/10 7:51:28
SHA 1: 72b94a8334d5539420f5df76666584f9f9e6fa83
MD5: 839fdc0ac889fa7c92339ae4c711e990
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main187.txt
Job Status: In process.
Commonly Registered Type: crypt user: main187
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:05:40
File Modified: 7/13/10 7:51:56
SHA 1: 2e8e48b0c0fe8ef6c037cedeea5efa4a208e97c4
MD5: b607f17e88030c1ac22b22b409fb750d
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main188.txt
Job Status: In process.
Commonly Registered Type: crypt user: main188
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:05:52
File Modified: 7/13/10 7:52:42
SHA 1: 4a8b04ffca4c48d2698b7c84220450450fb36d08
MD5: c8a9b71ebc7762fe5b2c1bcf55d1a4d1
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main189.txt
Job Status: In process.
Commonly Registered Type: crypt user: main189
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:06:02
File Modified: 7/13/10 7:52:58

SHA 1: 1ebc440cc54ac9323d7f3727a1ded830bd6ac06f
MD5: 2b457cad76b9b7dcc19fe5e7674d42cf
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main190.txt
Job Status: Finished on 7/19/10 17:57:27
Commonly Registered Type: crypt user: main190
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:06:13
File Modified: 7/13/10 7:53:10
SHA 1: a2a7e30f90928f54712cf33287c15499f99baf60
MD5: 986f88481ae088d15ce6473690d1750a
Result Type:
Result: corefaced48
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main191.txt
Job Status: Finished on 7/14/10 18:03:35
Commonly Registered Type: crypt user: main191
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:06:25
File Modified: 7/13/10 7:53:24
SHA 1: a79f12be9b1d99d6a522216ae7cde1860199f817
MD5: b956940abc7223cd8d407392b9ba9d56
Result Type:
Result: #1neilah03
Description: Unknown
Password Type: Password
Where Found: ---
C:\Main Test\MainTest-passwdFiles\Block4\main192.txt
Job Status: In process.
Commonly Registered Type: crypt user: main192
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:06:37
File Modified: 7/13/10 7:53:38
SHA 1: 2c8ed468018b6a111cc47e43f3ba36cfc52342f0
MD5: 7eccaa010436b691e348f19db3ca390e
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main193.txt
Job Status: In process.
Commonly Registered Type: crypt user: main193
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:06:50

File Modified: 7/13/10 7:55:34
SHA 1: 50844d29f3d72ac8398e6e35518bebc9d514994f
MD5: 7831dda1cf3534df6d2a487c2bfa84c1
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main194.txt
Job Status: In process.
Commonly Registered Type: crypt user: main194
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:07:04
File Modified: 7/13/10 7:55:56
SHA 1: fcce93a71eedb0e73c6f2994a85ff2c05cdde665
MD5: 2f72d8328aef66ea84e12b6d2d413842
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main195.txt
Job Status: In process.
Commonly Registered Type: crypt user: main195
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:07:15
File Modified: 7/13/10 7:56:10
SHA 1: 657ad301ab7c5bbae98eeb0b6c378bfe45fe50f6
MD5: 0e70f6134922a758622832486c3e9ec5
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main196.txt
Job Status: In process.
Commonly Registered Type: crypt user: main196
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:07:27
File Modified: 7/13/10 7:56:24
SHA 1: 7fcd163aaddba737bba8daf45578008883c98c37
MD5: 89ae78d9527c853d5893a33692ae2a6b
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main197.txt
Job Status: In process.
Commonly Registered Type: crypt user: main197
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:07:38
File Modified: 7/13/10 7:56:40
SHA 1: 962bf6bdbf26d59e5c8c3504f9c7aeb82836b4aa
MD5: b8358f5465d5736221a74f427d470ef8
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main198.txt
Job Status: In process.

Commonly Registered Type: crypt user: main198
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:07:49
File Modified: 7/13/10 7:56:52
SHA 1: 2f63dcfffd29b99e86cb270d6f714705567063a0
MD5: 2bb629431cfe0c0a519d2b5b1f7d8f6f
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main199.txt
Job Status: In process.
Commonly Registered Type: crypt user: main199
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:08:01
File Modified: 7/13/10 7:57:10
SHA 1: 38b0262242c374c204aa831b342649285c72036d
MD5: 32efb028c0d3b7a70c882e0189ac4483
No Password Found
C:\Main Test\MainTest-passwdFiles\Block4\main200.txt
Job Status: In process.
Commonly Registered Type: crypt user: main200
Identified Type: *nix passwd
File Size: 61
File Version: Unknown
Job Started: 7/14/10 9:08:11
File Modified: 7/13/10 7:57:22
SHA 1: 3f0fb6e4ddc7b14d6aab53d2c07f713d094d5aa5
MD5: bd208d755291359ef7ab35690a503f59
- No Password Found

APPENDIX 4: Time Analysis for all Blocks

The time analysis was performed on all the blocks, as described in section 4.3. The tables below show the time analysis done for blocks 1, 2, 3, and 4.

BLOCK # 1				
USERNAME	**START TIME**	**FINISH TIME**	**TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)**	**TIME TAKEN TO CRACK (IN SECONDS)**
main1	7/13/10 8:45:09	7/13/10 8:45:11	00:00:00:02	2
main2	7/13/10 8:45:20	7/13/10 8:45:23	00:00:00:03	3
main3	7/13/10 8:45:32	7/13/10 8:45:35	00:00:00:03	3
main4	7/13/10 8:45:43	7/13/10 8:45:47	00:00:00:04	4
main5	7/13/10 8:45:54	7/13/10 8:45:57	00:00:00:03	3
main6	7/13/10 8:46:06	7/13/10 8:46:08	00:00:00:02	2
main7	7/13/10 8:46:16	7/13/10 8:46:20	00:00:00:04	4
main8	7/13/10 8:46:27	7/13/10 8:46:31	00:00:00:04	4
main9	7/13/10 8:46:37	7/13/10 8:46:41	00:00:00:04	4
main10	7/13/10 8:46:48	7/13/10 8:46:53	00:00:00:05	5
main11	7/13/10 8:47:01	7/13/10 8:47:07	00:00:00:06	6
main12	7/13/10 8:47:11	7/13/10 8:47:19	00:00:00:08	8
main13	7/13/10 8:47:20	7/13/10 8:47:33	00:00:00:13	13
main14	7/13/10 8:47:30	7/13/10 8:47:47	00:00:00:17	17
main15	7/13/10 8:47:40	7/13/10 8:47:48	00:00:00:08	8
main16	7/13/10 8:47:49	7/13/10 8:47:52	00:00:00:03	3
main17	7/13/10 8:47:59	7/13/10 8:48:02	00:00:00:03	3
main18	7/13/10 8:48:09	7/13/10 8:48:21	00:00:00:12	12
main19	7/13/10 8:48:18	7/13/10 8:48:41	00:00:00:23	23
main20	7/13/10 8:48:28	7/13/10 8:48:28	00:00:00:00	0
main21	7/13/10 8:48:37	7/13/10 8:49:36	00:00:00:59	59
main22	7/13/10 8:48:53	7/13/10 8:49:28	00:00:00:35	35
main23	7/13/10 8:49:03	7/13/10 8:49:06	00:00:00:03	3
main24	7/13/10 8:49:13	7/13/10 8:50:29	00:00:01:16	76
main25	7/13/10 8:49:24	7/13/10 8:50:07	00:00:00:43	43
main26	7/13/10 8:49:34	7/13/10 8:50:50	00:00:01:16	76
main27	7/13/10 8:49:43	7/13/10 8:49:43	00:00:00:00	0
main28	7/13/10 8:49:52	7/13/10 8:50:30	00:00:00:38	38

main29	7/13/10 8:50:02	7/13/10 8:50:46	00:00:00:44	44
main30	7/13/10 8:50:12	7/13/10 8:50:45	00:00:00:33	33
main31	7/13/10 8:50:21	7/13/10 8:50:45	00:00:00:24	24
main32	7/13/10 8:50:31	7/13/10 8:51:09	00:00:00:38	38
main33	7/13/10 8:50:42	7/13/10 8:51:33	00:00:00:51	51
main34	7/13/10 8:50:52	7/13/10 8:52:15	00:00:01:23	83
main35	7/13/10 8:51:02	7/13/10 8:53:00	00:00:01:58	118
main36	7/13/10 8:51:12	7/13/10 8:52:37	00:00:01:25	85
main37	7/13/10 8:51:22	7/13/10 8:52:43	00:00:01:21	81
main38	7/13/10 8:51:36	7/13/10 8:53:13	00:00:01:37	97
main39	7/13/10 8:51:45	7/13/10 8:53:58	00:00:02:13	133
main40	7/13/10 8:51:56	7/13/10 8:53:00	00:00:01:04	64
main41	7/13/10 8:52:06	7/13/10 8:54:47	00:00:02:41	161
main42	7/13/10 8:52:16	7/13/10 8:54:06	00:00:01:50	110
main43	7/13/10 8:52:27	7/13/10 8:53:57	00:00:01:30	90
main44	7/13/10 8:52:38	7/13/10 8:54:36	00:00:01:58	118
main45	7/13/10 8:52:48	7/13/10 8:57:09	00:00:04:21	261
main46	7/13/10 8:52:58	7/13/10 8:56:11	00:00:03:13	193
main47	7/13/10 8:53:08	7/13/10 8:56:46	00:00:03:38	218
main48	7/13/10 8:53:18	7/13/10 8:55:46	00:00:02:28	148
main49	7/13/10 8:53:30	7/13/10 8:55:09	00:00:01:39	99
main50	7/13/10 8:53:39	7/13/10 8:56:39	00:00:03:00	180

BLOCK#2					
USERNAME	**START TIME**	**FINISH TIME**	**TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)**	**TIME TAKEN TO CRACK (IN SECONDS)**	**TIME TAKEN TO CRACK (IN MINUTES)**
main51	7/13/10 8:58:10	7/13/10 8:59:04	00:00:00:54	54	0.9
main52	7/13/10 8:58:22	7/13/10 8:59:23	00:00:01:01	61	1.016666667
main53	7/13/10 8:58:32	7/13/10 9:01:31	00:00:02:59	179	2.983333333
main54	7/13/10 8:58:43	7/13/10 10:26:06	00:01:27:23	5,243	87.38333333
main55	7/13/10 8:58:53	7/13/10 9:06:53	00:00:08:00	480	8
main56	7/13/10 8:59:03	7/13/10 9:12:57	00:00:13:54	834	13.9
main57	7/13/10 8:59:12	7/13/10 9:06:04	00:00:06:52	412	6.866666667
main58	7/13/10 8:59:27	7/13/10 9:50:06	00:00:50:39	3,039	50.65
main59	7/13/10 8:59:37	7/13/10 10:21:44	00:01:22:07	4,927	82.11666667

main60	7/13/10 8:59:47	7/13/10 9:06:20	00:00:06:33	393	6.55
main61	7/13/10 8:59:56	7/13/10 9:07:51	00:00:07:55	475	7.916666667
main62	7/13/10 9:00:06	7/13/10 9:46:34	00:00:46:28	2,788	46.46666667
main63	7/13/10 9:00:18	7/13/10 9:07:31	00:00:07:13	433	7.216666667
main64	7/13/10 9:00:29	7/13/10 10:25:01	00:01:24:32	5,072	84.53333333
main65	7/13/10 9:00:39	7/13/10 9:02:57	00:00:02:18	138	2.3
main66	7/13/10 9:00:50	7/13/10 9:25:51	00:00:25:01	1,501	25.01666667
main67	7/13/10 9:00:59	7/13/10 9:31:44	00:00:30:45	1,845	30.75
main68	7/13/10 9:01:10	7/13/10 10:02:28	00:01:01:18	3,678	61.3
main69	7/13/10 9:01:19	7/13/10 10:10:54	00:01:09:35	4,175	69.58333333
main70	7/13/10 9:01:29	7/13/10 10:03:46	00:01:02:17	3,737	62.28333333
main71	7/13/10 9:01:39	7/13/10 10:17:39	00:01:16:00	4,560	76
main72	7/13/10 9:01:49	7/13/10 10:37:21	00:01:35:32	5,732	95.53333333
main73	7/13/10 9:01:58	7/13/10 10:31:05	00:01:29:07	5,347	89.11666667
main74	7/13/10 9:02:08	7/13/10 10:11:19	00:01:09:11	4,151	69.18333333
main75	7/13/10 9:02:19	7/13/10 10:36:40	00:01:34:21	5,661	94.35
main76	7/13/10 9:02:28	7/13/10 10:31:01	00:01:28:33	5,313	88.55
main77	7/13/10 9:02:40	7/13/10 9:56:10	00:00:53:30	3,210	53.5
main78	7/13/10 9:02:50	7/13/10 10:17:15	00:01:14:25	4,465	74.41666667
main79	7/13/10 9:03:02	7/13/10 10:38:08	00:01:35:06	5,706	95.1
main80	7/13/10 9:03:12	7/13/10 10:37:16	00:01:34:04	5,644	94.06666667
main81	7/13/10 9:03:23	7/13/10 10:12:46	00:01:09:23	4,163	69.38333333
main82	7/13/10 9:03:32	7/13/10 10:42:44	00:01:39:12	5,952	99.2
main83	7/13/10 9:03:43	7/13/10 10:31:07	00:01:27:24	5,244	87.4
main84	7/13/10 9:03:55	7/13/10 10:25:48	00:01:21:53	4,913	81.88333333
main85	7/13/10 9:04:47	7/13/10 10:27:51	00:01:23:04	4,984	83.06666667
main87	7/13/10 9:07:48	7/13/10 10:43:50	00:01:36:02	5,762	96.03333333
main88	7/13/10 9:08:01	7/13/10 10:37:47	00:01:29:46	5,386	89.76666667
main89	7/13/10 9:08:12	7/13/10 10:49:36	00:01:41:24	6,084	101.4
main90	7/13/10 9:08:23	7/13/10 10:42:37	00:01:34:14	5,654	94.23333333
main91	7/13/10 9:08:34	7/13/10 10:57:53	00:01:49:19	6,559	109.3166667
main92	7/13/10 9:08:46	7/13/10 10:48:38	00:01:39:52	5,992	99.86666667
main93	7/13/10 9:08:57	7/13/10 11:02:30	00:01:53:33	6,813	113.55
main94	7/13/10 9:09:06	7/13/10 10:40:27	00:01:31:21	5,481	91.35
main95	7/13/10 9:09:16	7/13/10 10:53:58	00:01:44:42	6,282	104.7
main96	7/13/10 9:09:29	7/13/10 10:51:14	00:01:41:45	6,105	101.75

main97	7/13/10 9:09:48	7/13/10 10:37:49	00:01:28:01	5,281	88.01666667
main98	7/13/10 9:10:07	7/13/10 11:02:58	00:01:52:51	6,771	112.85
main99	7/13/10 9:10:19	7/13/10 11:00:43	00:01:50:24	6,624	110.4
main100	7/13/10 9:10:29	7/13/10 10:56:13	00:01:45:44	6,344	105.7333333

BLOCK#3						
USERNAME	**START TIME**	**FINISH TIME**	**TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)**	**TIME TAKEN TO CRACK (IN SECONDS)**	**TIME TAKEN TO CRACK (IN MINUTES)**	**TIME TAKEN TO CRACK (IN HOURS)**
main101	7/13/10 11:31:48	7/13/10 13:21:33	00:01:49:45	6,585	109.75	1.829166667
main102	7/13/10 11:32:04	7/13/10 16:16:20	00:04:44:16	17,056	284.2666667	4.737777778
main103	7/13/10 11:32:15	7/13/10 16:35:42	00:05:03:27	18,207	303.45	5.0575
main104	7/13/10 11:32:27	7/13/10 18:45:03	00:07:12:36	25,956	432.6	7.21
main105	7/13/10 11:32:42	7/13/10 19:01:19	00:07:28:37	26,917	448.6166667	7.476944444
main106	7/13/10 11:32:53	7/13/10 18:09:00	00:06:36:07	23,767	396.1166667	6.601944444
main107	7/13/10 11:33:05	7/13/10 15:37:26	00:04:04:21	14,661	244.35	4.0725
main108	7/13/10 11:33:16	7/13/10 16:36:11	00:05:02:55	18,175	302.9166667	5.048611111
main109	7/13/10 11:33:30	7/13/10 20:01:49	00:08:28:19	30,499	508.3166667	8.471944444
main110	7/13/10 11:33:41	7/13/10 17:23:28	00:05:49:47	20,987	349.7833333	5.829722222
main111	7/13/10 11:33:57	7/13/10 20:15:10	00:08:41:13	31,273	521.2166667	8.686944444
main112	7/13/10 11:34:09	7/13/10 16:47:50	00:05:13:41	18,821	313.6833333	5.228055556
main113	7/13/10 11:34:22	7/13/10 13:55:28	00:02:21:06	8,466	141.1	2.351666667
main114	7/13/10 11:34:34	7/13/10 20:29:53	00:08:55:19	32,119	535.3166667	8.921944444
main115	7/13/10 11:34:46	7/13/10 12:07:47	00:00:33:01	1,981	33.01666667	0.550277778
main116	7/13/10 11:34:59	7/13/10 20:09:49	00:08:34:50	30,890	514.8333333	8.580555556
main117	7/13/10 11:35:10	7/13/10 19:20:38	00:07:45:28	27,928	465.4666667	7.757777778
main118	7/13/10 11:35:21	7/13/10 19:45:23	00:08:10:02	29,402	490.0333333	8.167222222
main119	7/13/10 11:35:31	7/13/10 21:03:21	00:09:27:50	34,070	567.8333333	9.463888889
main120	7/13/10 11:35:45	7/13/10 15:43:33	00:04:07:48	14,868	247.8	4.13
main121	7/13/10 11:35:55	7/13/10 19:41:03	00:08:05:08	29,108	485.1333333	8.085555556
main122	7/13/10 11:36:07	7/13/10 17:41:01	00:06:04:54	21,894	364.9	6.081666667
main123	7/13/10 11:36:19	7/13/10 22:24:25	00:10:48:06	38,886	648.1	10.80166667
main124	7/13/10 11:36:31	7/13/10 19:46:13	00:08:09:42	29,382	489.7	8.161666667
main125	7/13/10 11:36:42	7/13/10 18:41:47	00:07:05:05	25,505	425.0833333	7.084722222
main126	7/13/10 11:36:54	7/13/10 22:03:42	00:10:26:48	37,608	626.8	10.44666667

main127	7/13/10 11:37:06	7/13/10 22:13:36	00:10:36:30	38,190	636.5	10.60833333
main128	7/13/10 11:37:19	7/13/10 22:14:53	00:10:37:34	38,254	637.5666667	10.62611111
main129	7/13/10 11:37:30	7/13/10 22:28:31	00:10:51:01	39,061	651.0166667	10.85027778
main130	7/13/10 11:37:41	7/13/10 22:04:45	00:10:27:04	37,624	627.0666667	10.45111111
main131	7/13/10 11:37:50	7/13/10 18:42:20	00:07:04:30	25,470	424.5	7.075
main132	7/13/10 11:38:00	7/13/10 19:58:03	00:08:20:03	30,003	500.05	8.334166667
main133	7/13/10 11:38:10	7/13/10 14:51:45	00:03:13:35	11,615	193.5833333	3.226388889
main134	7/13/10 11:38:24	7/13/10 18:51:07	00:07:12:43	25,963	432.7166667	7.211944444
main135	7/13/10 11:38:34	7/13/10 22:24:32	00:10:45:58	38,758	645.9666667	10.76611111
main136	7/13/10 11:38:48	7/13/10 19:15:55	00:07:37:07	27,427	457.1166667	7.618611111
main137	7/13/10 11:39:00	7/13/10 22:50:06	00:11:11:06	40,266	671.1	11.185
main138	7/13/10 11:39:13	7/13/10 22:51:47	00:11:12:34	40,354	672.5666667	11.20944444
main139	7/13/10 11:39:27	7/13/10 22:51:51	00:11:12:24	40,344	672.4	11.20666667
main140	7/13/10 11:39:37	7/13/10 15:58:10	00:04:18:33	15,513	258.55	4.309166667
main141	7/13/10 11:44:19	7/13/10 22:13:13	00:10:28:54	37,734	628.9	10.48166667
main142	7/13/10 11:40:10	7/13/10 19:43:53	00:08:03:43	29,023	483.7166667	8.061944444
main143	7/13/10 11:40:21	7/13/10 21:16:37	00:09:36:16	34,576	576.2666667	9.604444444
main144	7/13/10 11:40:34	7/13/10 21:45:48	00:10:05:14	36,314	605.2333333	10.08722222
main145	7/13/10 11:40:49	7/13/10 22:53:04	00:11:12:15	40,335	672.25	11.20416667
main146	7/13/10 11:41:00	7/13/10 19:17:31	00:07:36:31	27,391	456.5166667	7.608611111
main147	7/13/10 11:41:10	7/13/10 22:55:14	00:11:14:04	40,444	674.0666667	11.23444444
main148	7/13/10 11:41:22	7/13/10 23:20:56	00:11:39:34	41,974	699.5666667	11.65944444
main149	7/13/10 11:41:33	7/13/10 22:02:50	00:10:21:17	37,277	621.2833333	10.35472222
main150	7/13/10 11:41:45	7/13/10 23:36:09	00:11:54:24	42,864	714.4	11.90666667

BLOCK#4							
USERNAME	**START TIME**	**FINISH TIME**	**TIME TAKEN TO CRACK (FORMAT DD:HH:MM:SS)**	**TIME TAKEN TO CRACK (IN SECONDS)**	**TIME TAKEN TO CRACK (IN MINUTES)**	**TIME TAKEN TO CRACK (IN HOURS)**	**TIME TAKEN TO CRACK (IN DAYS)**
					0	0	
main151	7/14/10 8:58:56	7/15/10 3:25:21	00:18:26:25	66,385	1106.416667	18.44027778	0.768344907
main152	7/14/10 8:59:14	7/15/10 23:03:12	01:14:03:58	137,038	2283.966667	38.06611111	1.586087963
main153	7/14/10 8:59:25	7/15/10 2:07:12	00:17:07:47	61667	1027.783333	17.12972222	0.713738426
main154	7/14/10 8:59:36	7/17/10 8:02:59	02:23:03:23	255803	4263.383333	71.05638889	2.96068287
main155	7/14/10 8:59:47	7/17/10 15:41:19	03:06:41:32	283292	4721.533333	78.69222222	3.278842593
main156	7/14/10 8:59:59	7/17/10 7:14:26	02:22:14:27	252867	4214.45	70.24083333	2.926701389

main157	7/14/10 9:00:12	7/18/10 7:49:16	03:22:49:04	341344	5689.066667	94.8177777 8	3.950740741
main158	7/14/10 9:00:23	7/17/10 12:27:54	03:03:27:31	271651	4527.516667	75.4586111 1	3.144108796
main159	7/14/10 9:00:35	7/19/10 10:04:05	05:01:03:30	435810	7263.5	121.058333 3	5.044097222
main160	7/14/10 9:00:45	7/16/10 17:23:46	02:08:23:01	202981	3383.016667	56.3836111 1	2.34931713
main161	7/14/10 9:00:56	7/18/10 5:06:09	03:20:05:13	331513	5525.216667	92.0869444 4	3.836956019
main162	7/14/10 9:01:06	7/16/10 2:06:11	01:17:05:05	147905	2465.083333	41.0847222 2	1.711863426
main164	7/14/10 9:01:28	7/18/10 11:02:14	04:02:00:46	352846	5880.766667	98.0127777 8	4.083865741
main166	7/14/10 9:01:49	7/17/10 3:57:51	02:18:56:02	240962	4016.033333	66.9338888 9	2.788912037
main169	7/14/10 9:02:21	7/19/10 19:42:00	05:10:39:39	470379	7839.65	130.660833 3	5.444201389
main170	7/14/10 9:02:31	7/20/10 5:48:47	05:20:46:16	506776	8446.266667	140.771111 1	5.865462963
main172	7/14/10 9:02:53	7/20/10 5:30:36	05:20:27:43	505663	8427.716667	140.461944 4	5.852581019
main173	7/14/10 9:03:03	7/19/10 19:34:03	05:10:31:00	469860	7831	130.516666 7	5.438194444
main174	7/14/10 9:03:14	7/17/10 15:39:07	03:06:35:53	282953	4715.883333	78.5980555 6	3.274918981
main175	7/14/10 9:03:25	7/15/10 16:34:25	01:07:31:00	113460	1891	31.5166666 7	1.313194444
main176	7/14/10 9:03:36	7/19/10 17:58:48	05:08:55:12	464112	7735.2	128.92	5.371666667
main184	7/14/10 9:05:09	7/15/10 15:10:20	01:06:05:11	108311	1805.183333	30.0863888 9	1.253599537
main190	7/14/10 9:06:13	7/19/10 17:57:27	05:08:51:14	463874	7731.233333	128.853888 9	5.368912037
main191	7/14/10 9:06:25	7/14/10 18:03:35	00:08:57:10	32230	537.1666667	8.95277777 8	0.373032407
main163	7/14/10 9:01:18	In process.			0	0	0
main165	7/14/10 9:01:39	In process.			0	0	0
main167	7/14/10 9:02:00	In process.			0	0	0
main168	7/14/10 9:02:11	In process.			0	0	0
main171	7/14/10 9:02:42	In process.			0	0	0
main177	7/14/10 9:03:46	In process.			0	0	0
main178	7/14/10 9:03:57	In process.			0	0	0
main179	7/14/10 9:04:07	In process.			0	0	0
main180	7/14/10 9:04:19	In process.			0	0	0
main181	7/14/10 9:04:30	In process.			0	0	0
main182	7/14/10 9:04:41	In process.			0	0	0
main183	7/14/10 9:04:53	In process.			0	0	0
main185	7/14/10 9:05:19	In process.			0	0	0
main186	7/14/10 9:05:30	In process.			0	0	0
main187	7/14/10 9:05:40	In process.			0	0	0
main188	7/14/10 9:05:52	In process.			0	0	0
main189	7/14/10 9:06:02	In process.			0	0	0

main192	7/14/10 9:06:37	In process.			0	0	0
main193	7/14/10 9:06:50	In process.			0	0	0
main194	7/14/10 9:07:04	In process.			0	0	0
main195	7/14/10 9:07:15	In process.			0	0	0
main196	7/14/10 9:07:27	In process.			0	0	0
main197	7/14/10 9:07:38	In process.			0	0	0
main198	7/14/10 9:07:49	In process.			0	0	0
main199	7/14/10 9:08:01	In process.			0	0	0
main200	7/14/10 9:08:11	In process.			0	0	0